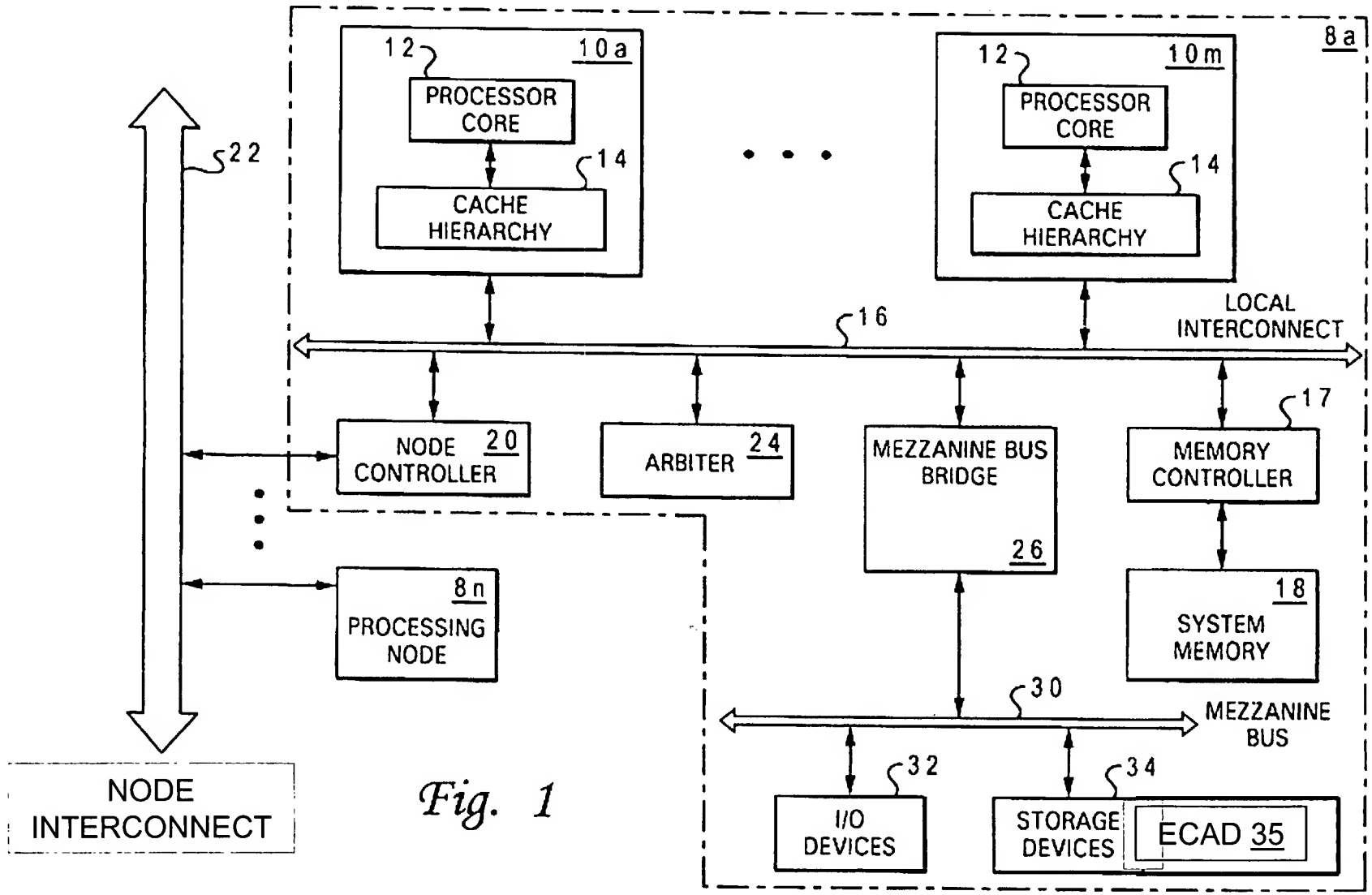
8a
12
10a
PROCESSOR CORE
14
CACHE HIERARCHY
12
10m
PROCESSOR CORE
14
CACHE HIERARCHY
22
16
LOCAL INTERCONNECT
NODE CONTROLLER 20
ARBITER 24
MEZZANINE BUS BRIDGE 26
17
MEMORY CONTROLLER
8n
PROCESSING NODE
18
SYSTEM MEMORY
30
MEZZANINE BUS
32
34
I/O DEVICES
STORAGE DEVICES
ECAD 35
NODE INTERCONNECT

Fig. 1

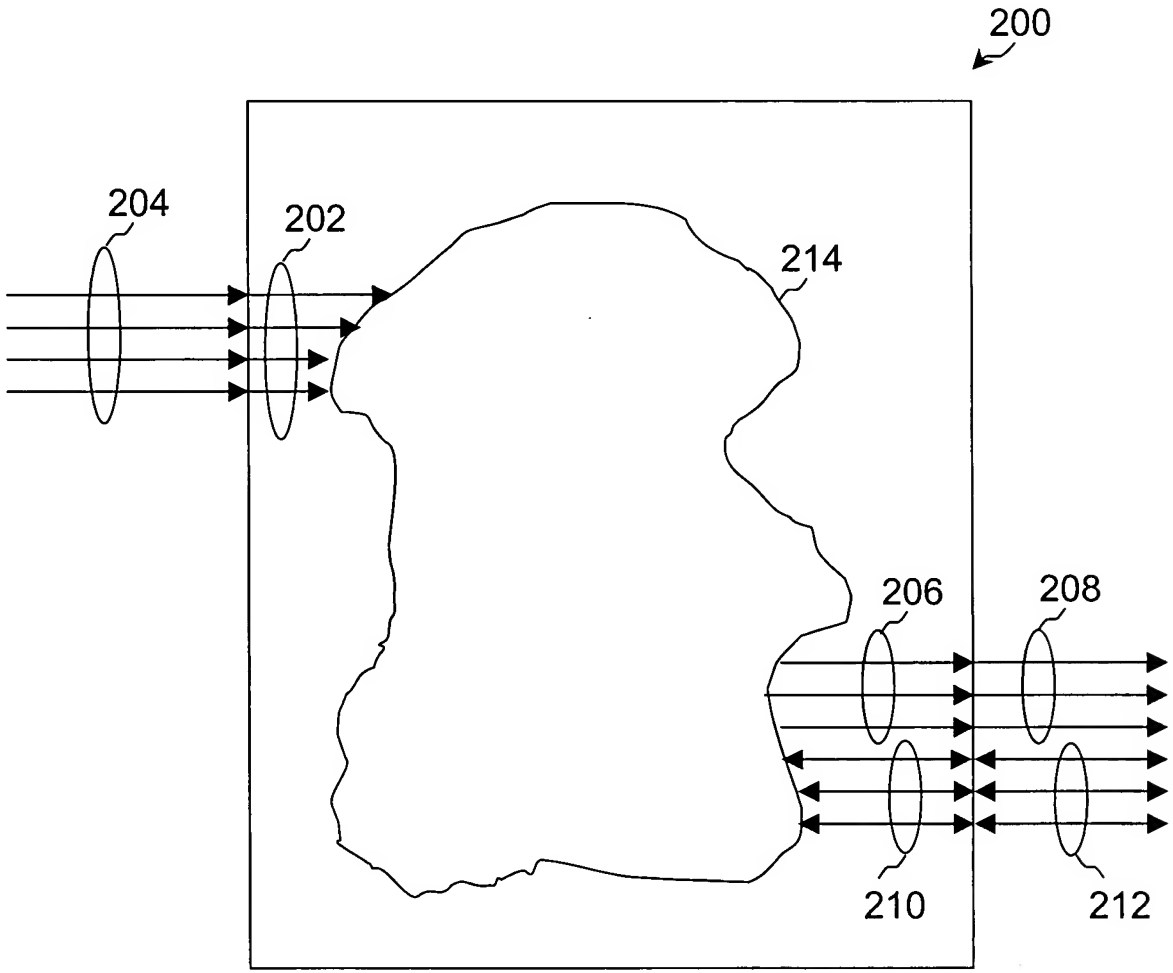

Figure 2

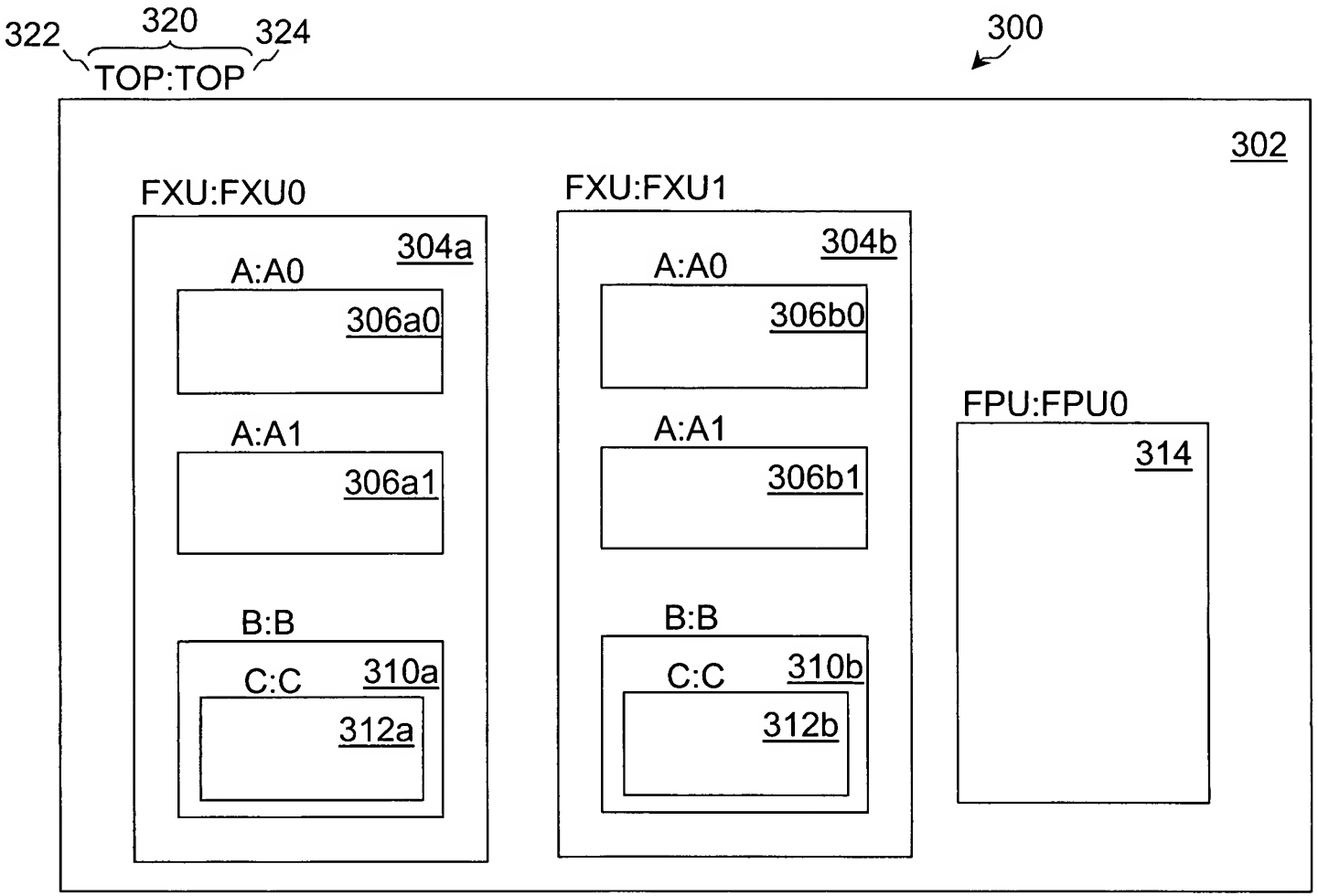
322
320
324
TOP:TOP
300
302
FXU:FXU0
304a
A:A0
306a0
A:A1
306a1
B:B
C:C
310a
312a
FXU:FXU1
304b
A:A0
306b0
A:A1
306b1
B:B
C:C
310b
312b
FPU:FPU0
314

Figure 3

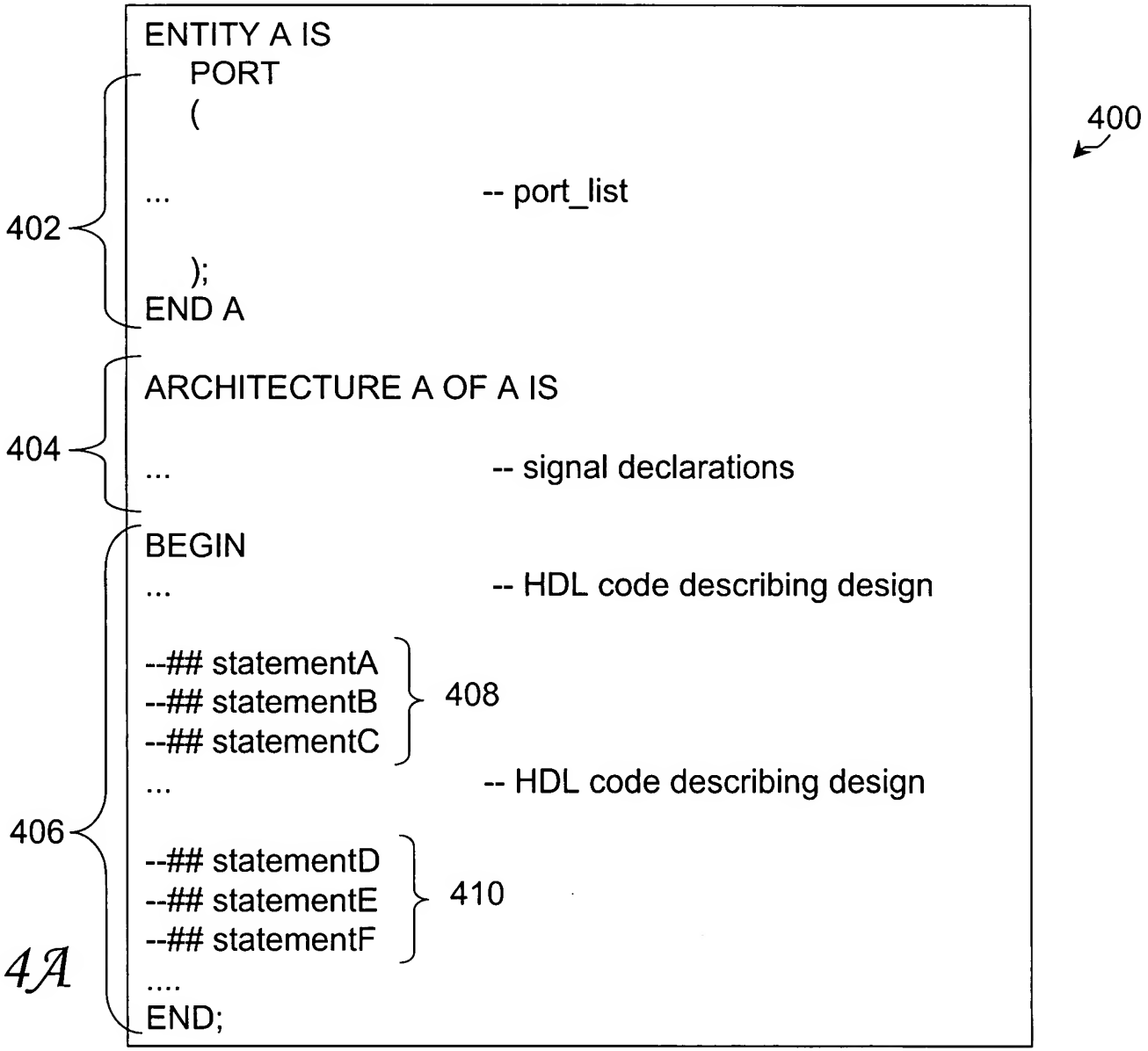
ENTITY A IS
PORT
(
...
-- port_list
);
END A
ARCHITECTURE A OF A IS
...
-- signal declarations
BEGIN
...
-- HDL code describing design
--## statementA
--## statementB
--## statementC
408
...
-- HDL code describing design
--## statementD
--## statementE
--## statementF
410
....
END;
400
402
404
406

Figure 4A

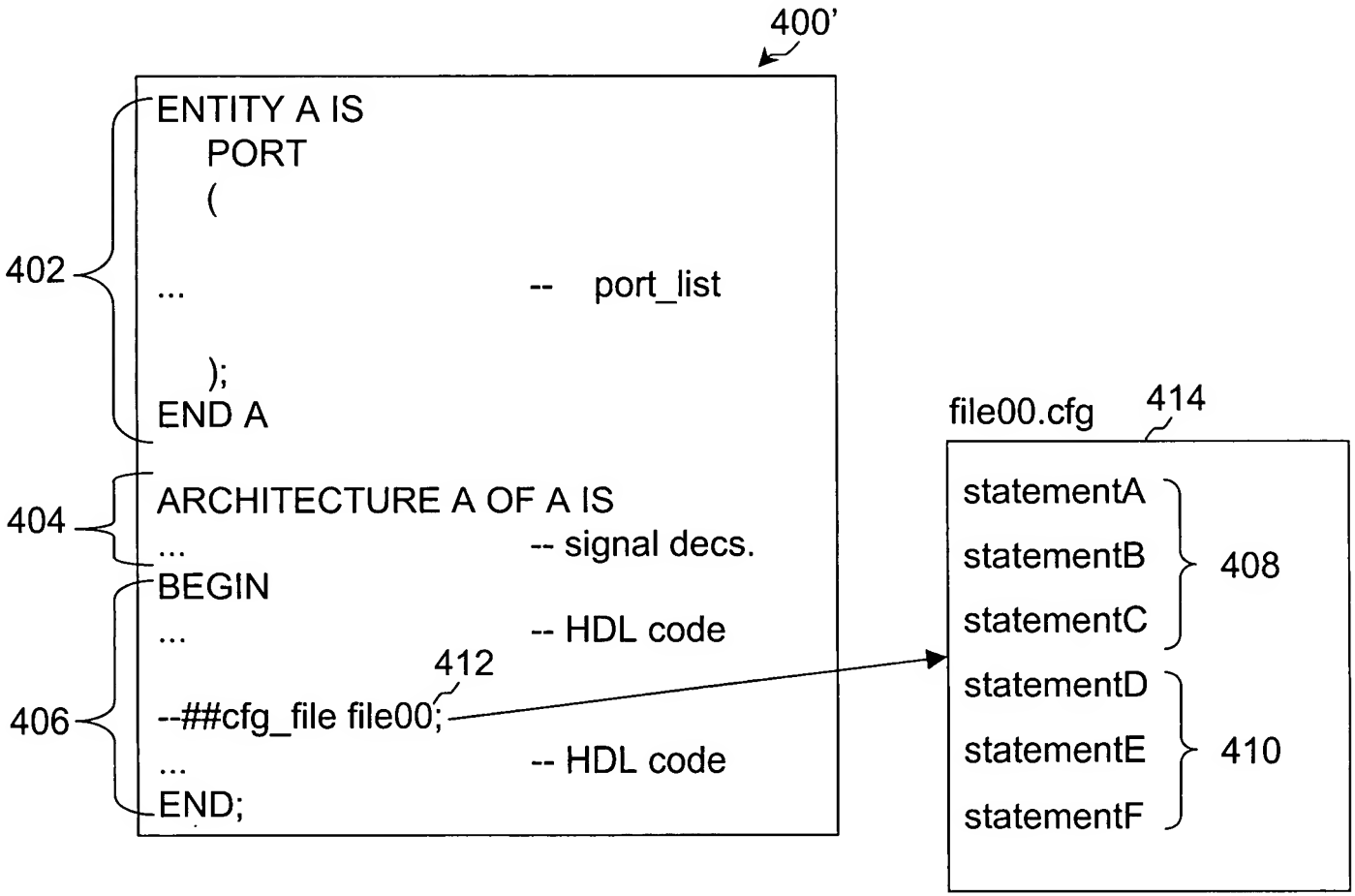
400'
ENTITY A IS
PORT
(
...
-- port_list
);
END A
ARCHITECTURE A OF A IS
...
-- signal decs.
BEGIN
...
-- HDL code
--##cfg_file file00;
...
-- HDL code
END;
402
404
406
412
file00.cfg
414
statementA
statementB
statementC
408
statementD
statementE
statementF
410

Figure 4B

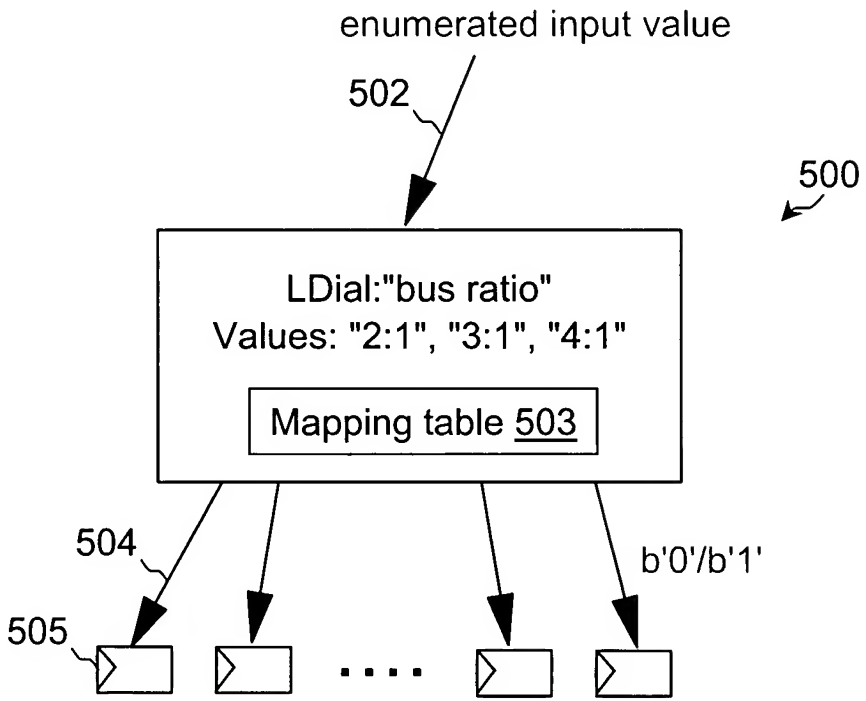
enumerated input value
502
500
LDial:"bus ratio"
Values: "2:1", "3:1", "4:1"
Mapping table 503
504
505
b'0'/b'1'

Figure 5A

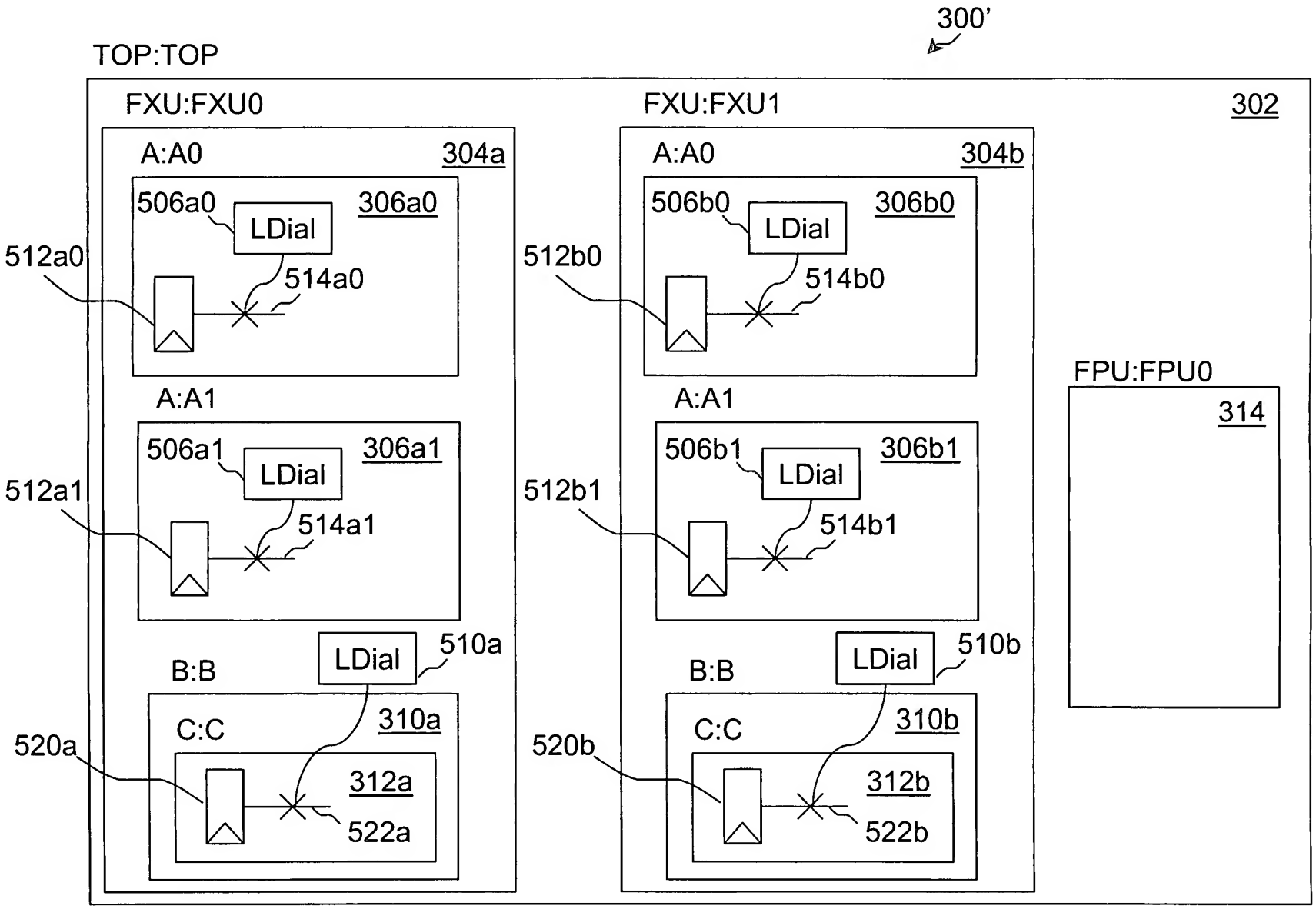
300'
TOP:TOP
FXU:FXU0
FXU:FXU1
302
A:A0
304a
A:A0
304b
506a0
LDial
306a0
506b0
LDial
306b0
512a0
514a0
512b0
514b0
A:A1
A:A1
FPU:FPU0
314
506a1
LDial
306a1
506b1
LDial
306b1
512a1
514a1
512b1
514b1
LDial
510a
LDial
510b
B:B
B:B
C:C
310a
C:C
310b
520a
312a
520b
312b
522a
522b

Figure 5B

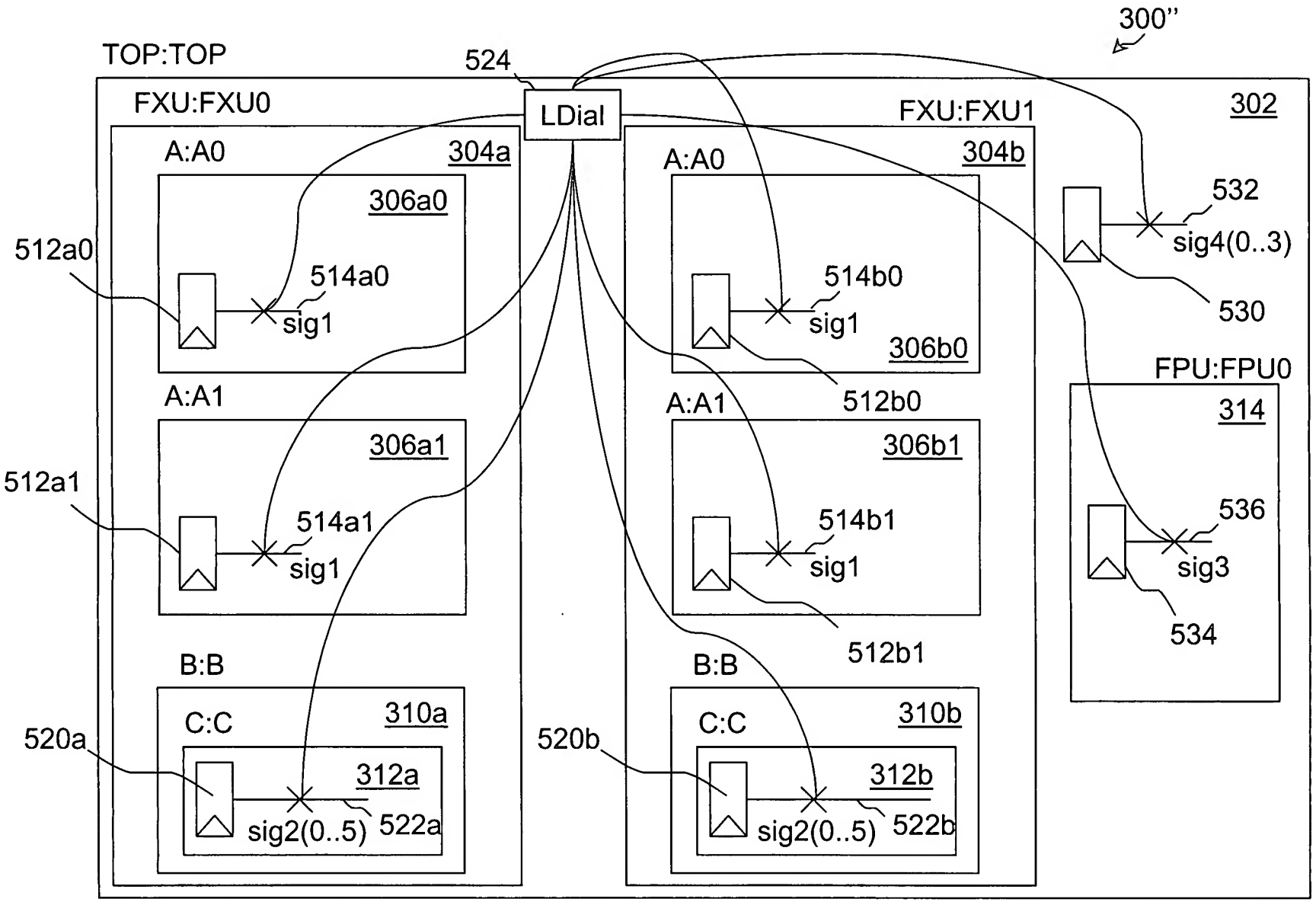
300"
TOP:TOP
524
LDial
FXU:FXU0
FXU:FXU1
302
A:A0
304a
306a0
512a0
514a0
sig1
A:A1
306a1
512a1
514a1
sig1
B:B
C:C
310a
520a
312a
sig2(0..5)
522a
A:A0
304b
306b0
514b0
sig1
512b0
A:A1
306b1
514b1
sig1
512b1
B:B
C:C
310b
520b
312b
sig2(0..5)
522b
532
sig4(0..3)
530
FPU:FPU0
314
536
sig3
534

Figure 5C

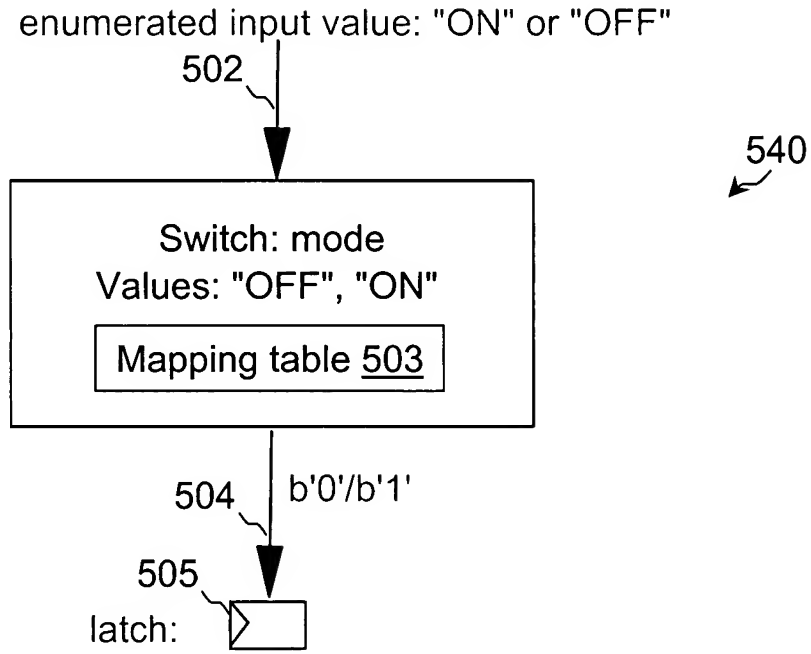
enumerated input value: "ON" or "OFF"
502
540
Switch: mode
Values: "OFF", "ON"
Mapping table 503
504
b'0'/b'1'
505
latch:

Figure 5D

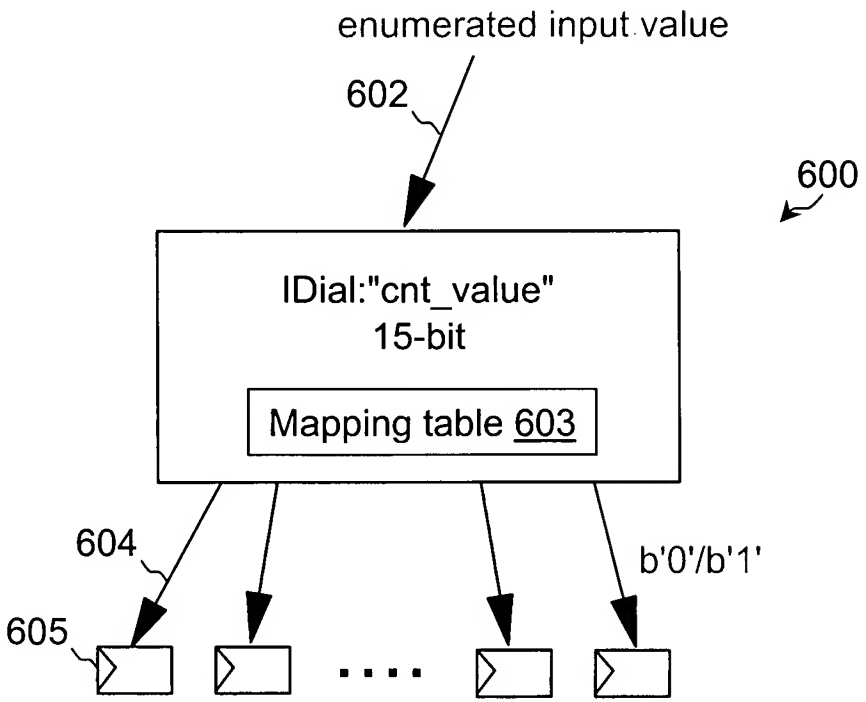
enumerated input value
602
600
IDial:"cnt_value"
15-bit
Mapping table 603
604
605
b'0'/b'1'

Figure 6A

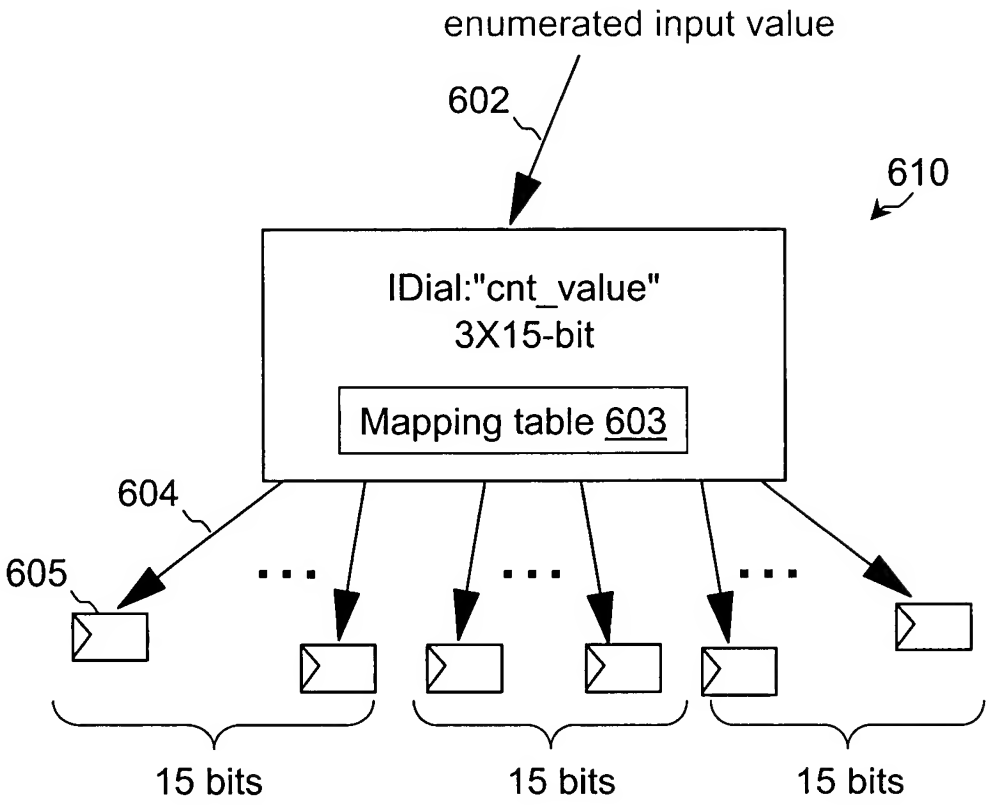
enumerated input value
602
610
IDial:"cnt_value"
3X15-bit
Mapping table 603
604
605
15 bits
15 bits
15 bits

Figure 6B

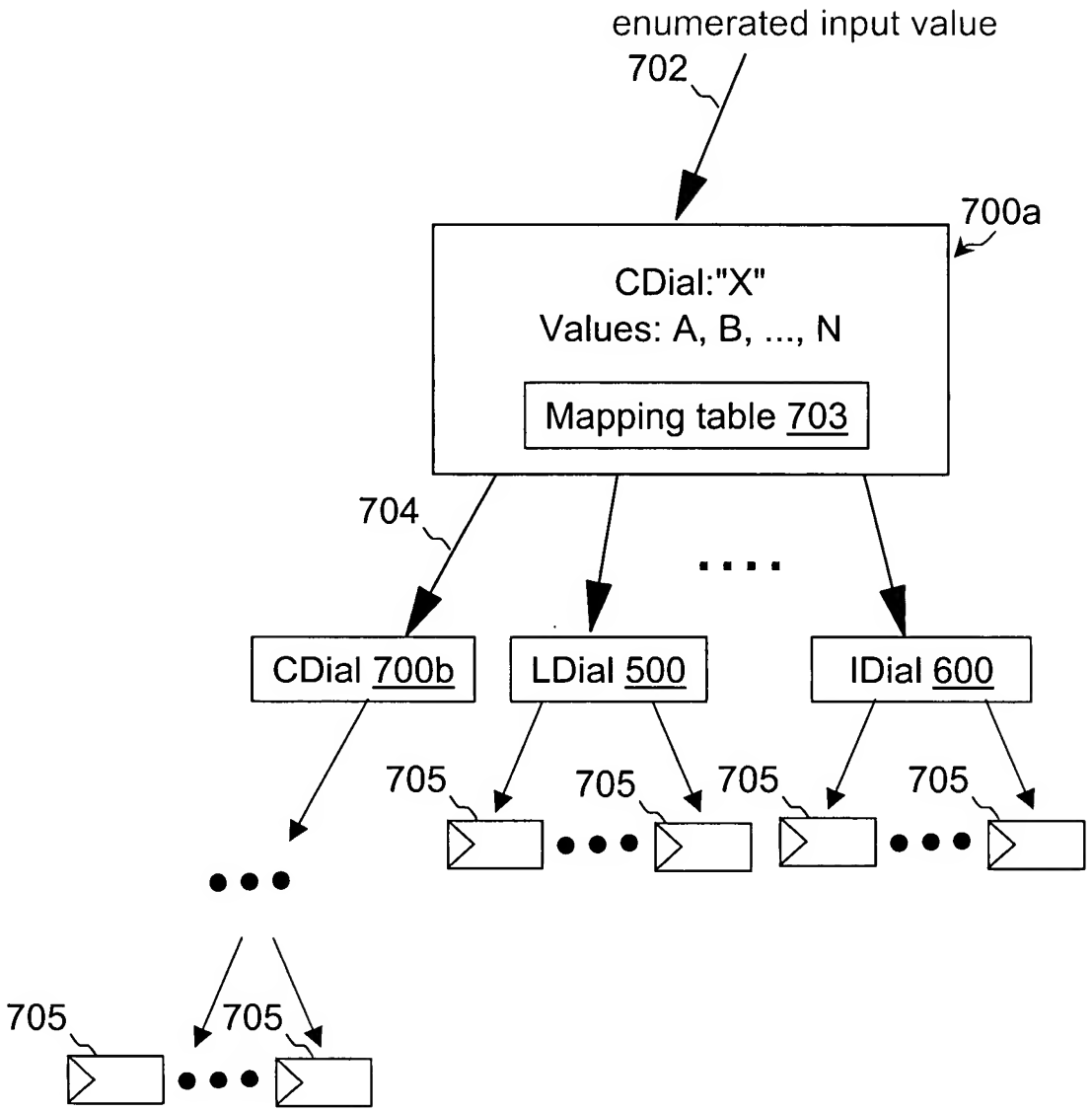
enumerated input value
702
700a
CDial:"X"
Values: A, B, ..., N
Mapping table 703
704
CDial 700b
LDial 500
IDial 600
705
705
705
705
705
705

Figure 7A

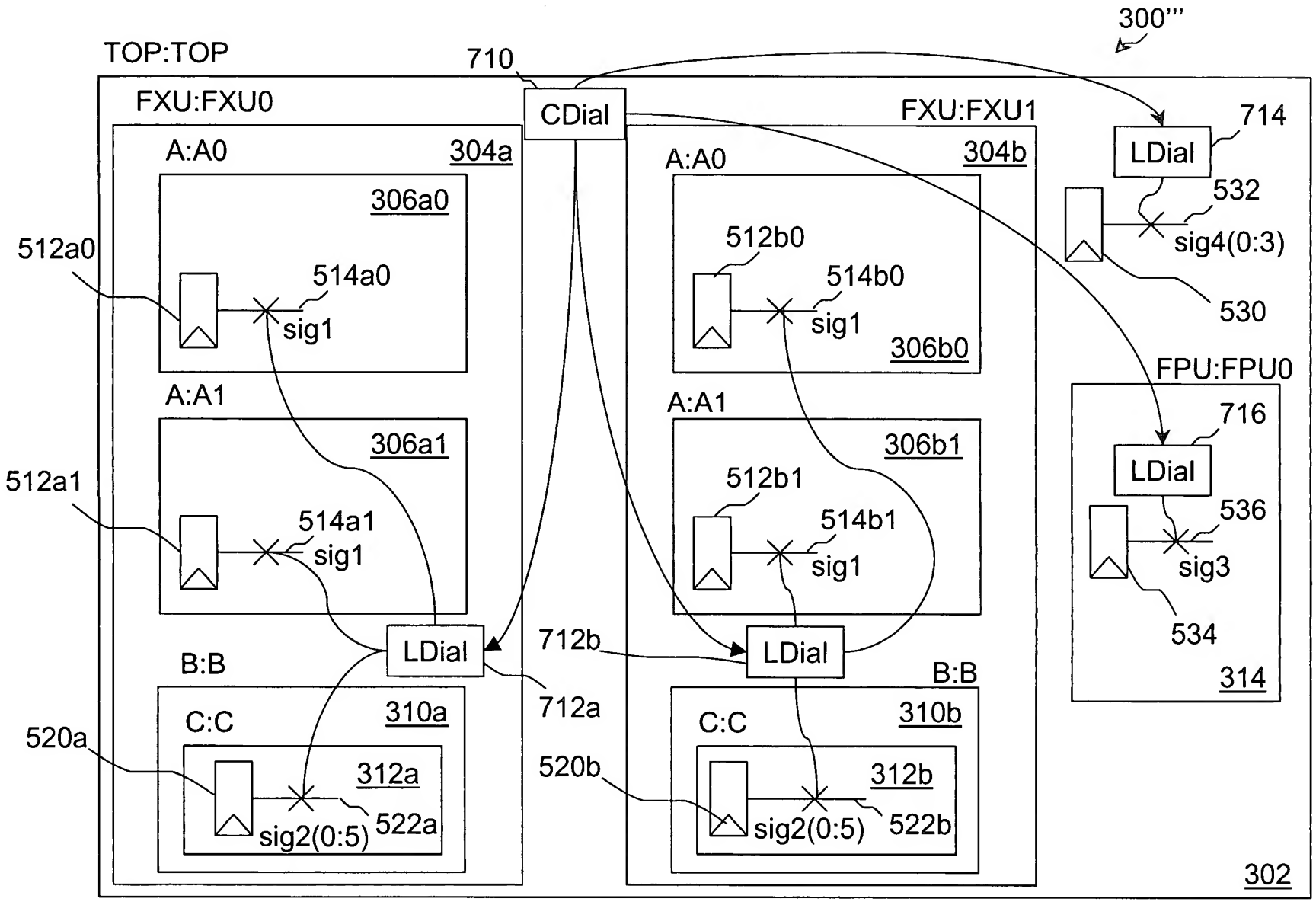
300'''
TOP:TOP
710
FXU:FXU0
CDial
FXU:FXU1
714
LDial
A:A0
304a
A:A0
304b
532
306a0
512b0
sig4(0:3)
512a0
514a0
514b0
530
sig1
sig1
306b0
FPU:FPU0
A:A1
A:A1
716
306a1
306b1
LDial
512a1
512b1
536
514a1
514b1
sig3
sig1
sig1
534
712b
LDial
LDial
B:B
B:B
314
712a
C:C
310a
C:C
310b
520a
312a
520b
312b
522a
sig2(0:5)
sig2(0:5)
522b
302

Figure 7B

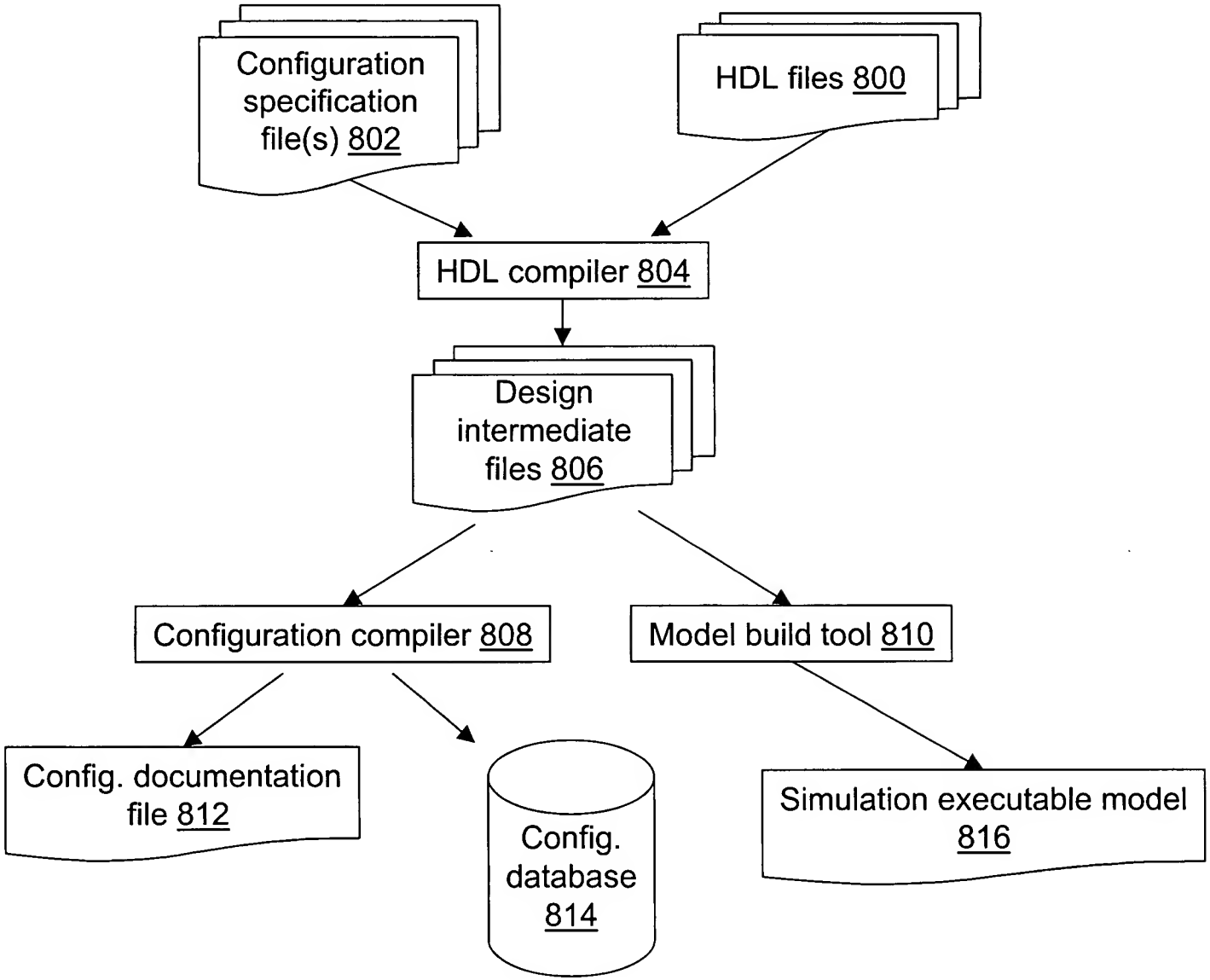
Configuration specification file(s) 802
HDL files 800
HDL compiler 804
Design intermediate files 806
Configuration compiler 808
Model build tool 810
Config. documentation file 812
Config. database 814
Simulation executable model 816

Figure 8

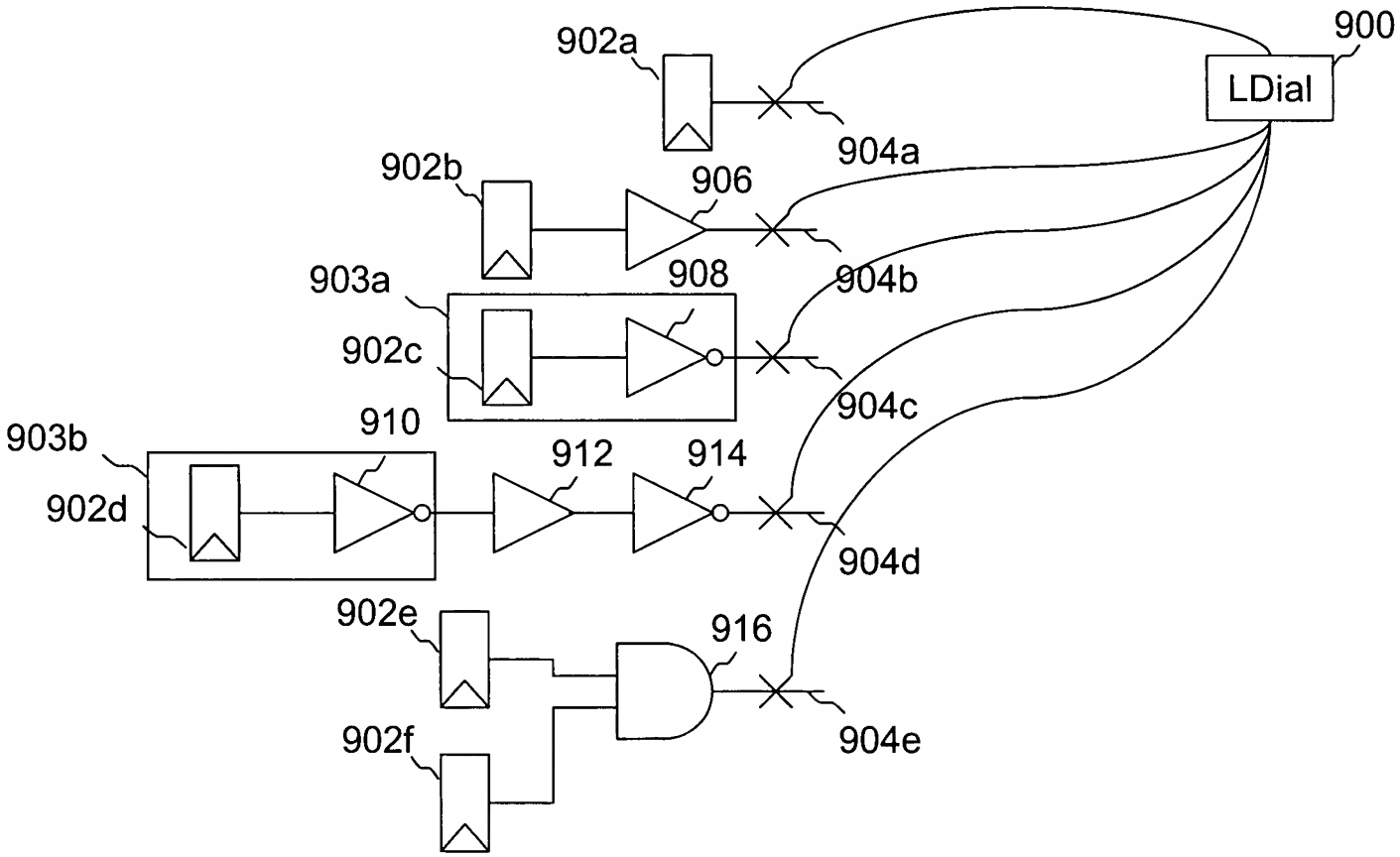
902a
900
LDial
904a
902b
906
908
903a
904b
902c
904c
910
903b
912
914
902d
904d
902e
916
902f
904e

Figure 9A

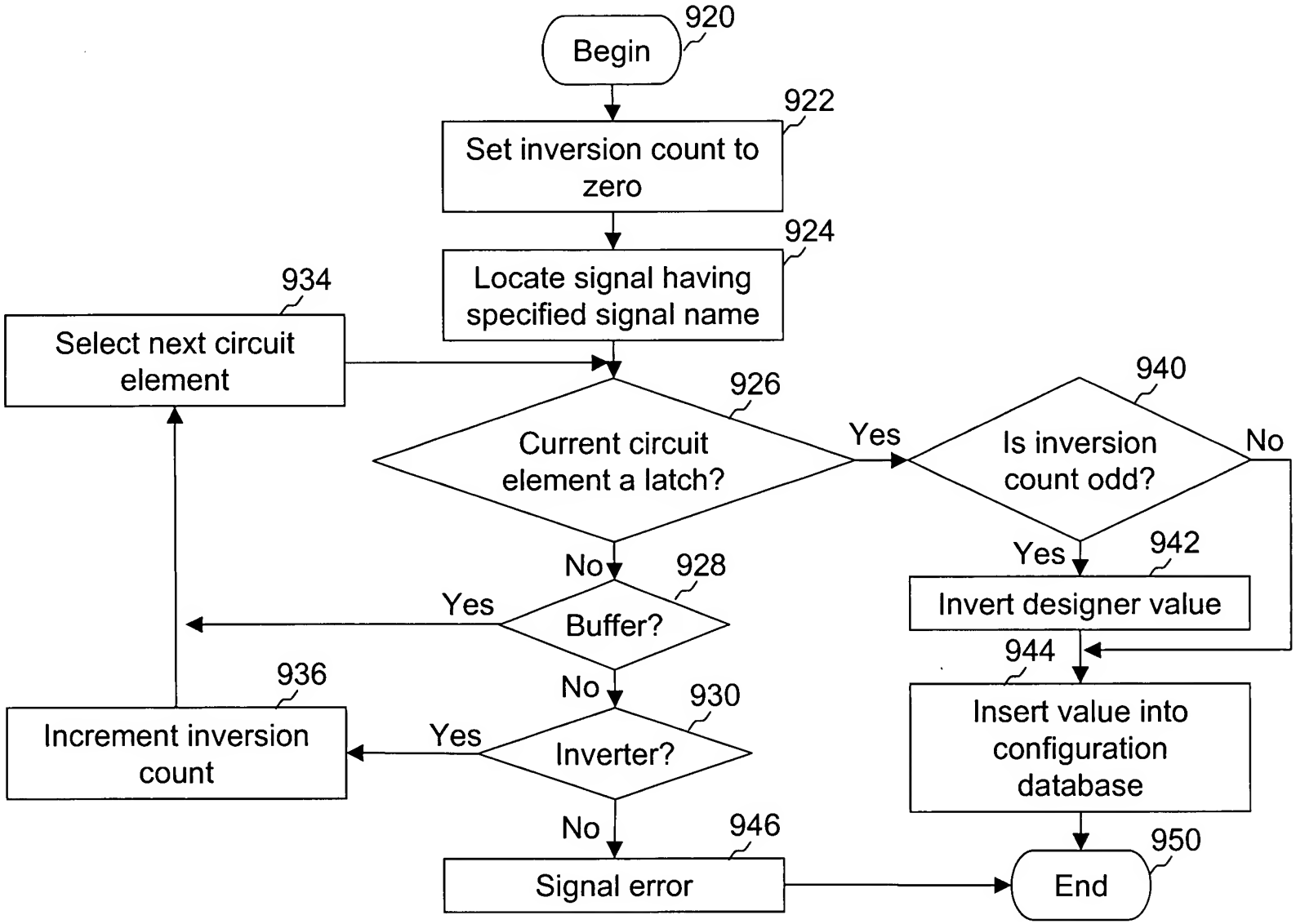
Begin
920
Set inversion count to zero
922
Locate signal having specified signal name
924
Select next circuit element
934
Current circuit element a latch?
926
Yes
Is inversion count odd?
940
No
Yes
942
Invert designer value
No
928
Yes
Buffer?
944
Insert value into configuration database
No
930
936
Increment inversion count
Yes
Inverter?
No
946
Signal error
End
950

Figure 9B

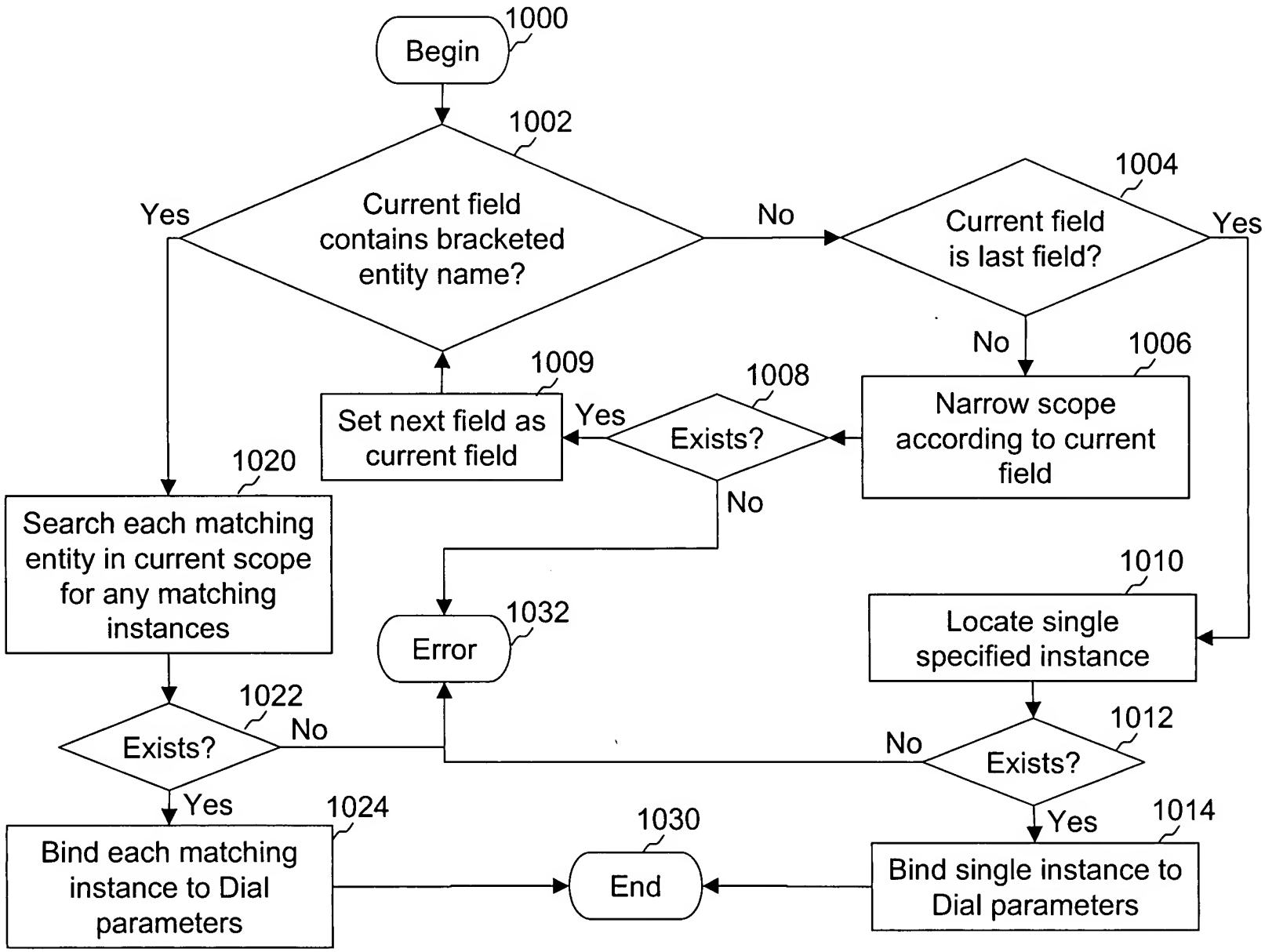
Begin
1000
1002
Current field contains bracketed entity name?
Yes
No
1004
Current field is last field?
Yes
No
1006
Narrow scope according to current field
1008
Exists?
Yes
No
1009
Set next field as current field
1020
Search each matching entity in current scope for any matching instances
1032
Error
1010
Locate single specified instance
1012
Exists?
No
Yes
1014
Bind single instance to Dial parameters
1022
Exists?
No
Yes
1024
Bind each matching instance to Dial parameters
1030
End

Figure 10

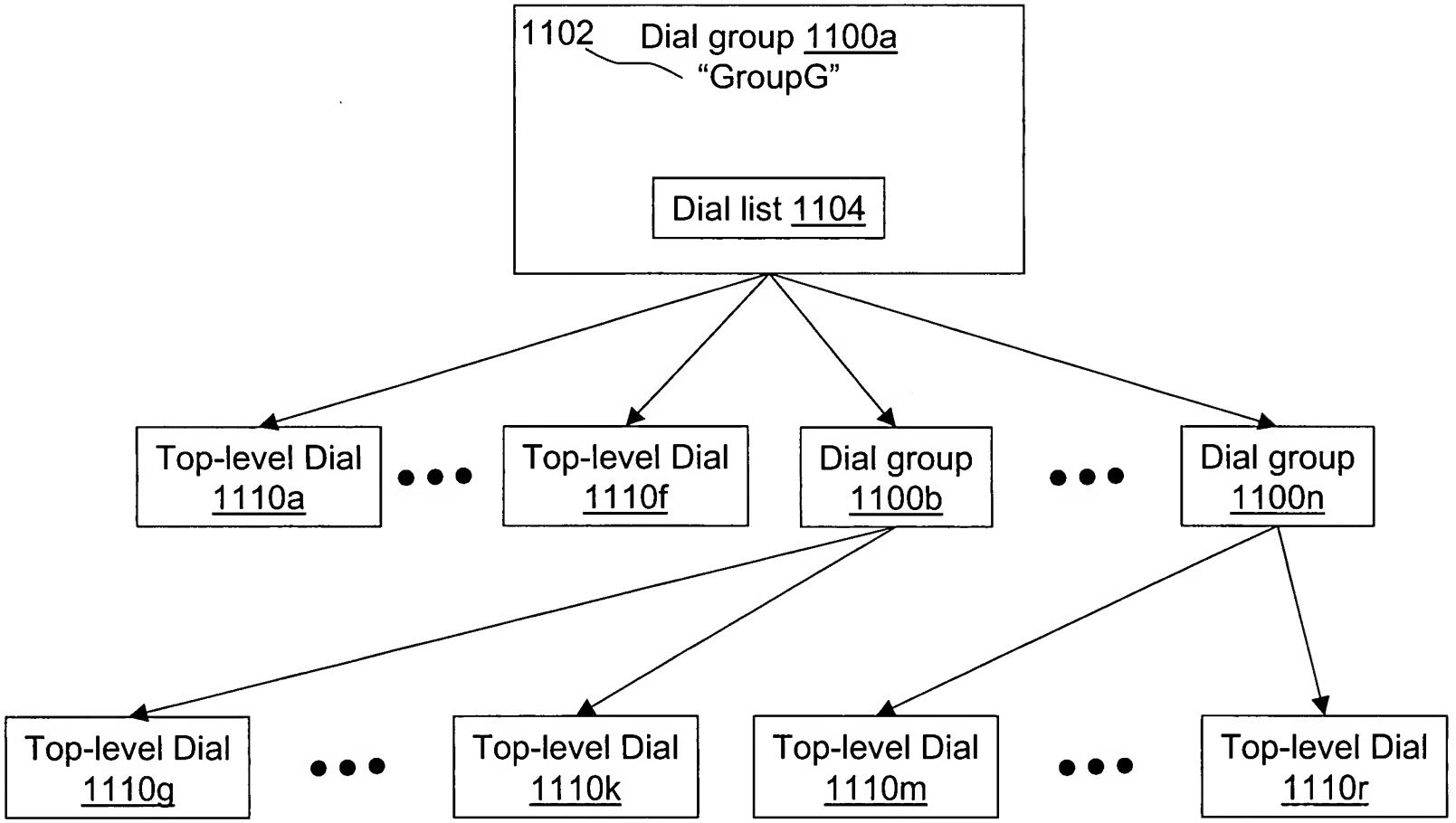
1102
Dial group 1100a
"GroupG"
Dial list 1104
Top-level Dial 1110a
Top-level Dial 1110f
Dial group 1100b
Dial group 1100n
Top-level Dial 1110g
Top-level Dial 1110k
Top-level Dial 1110m
Top-level Dial 1110r

Figure 11A

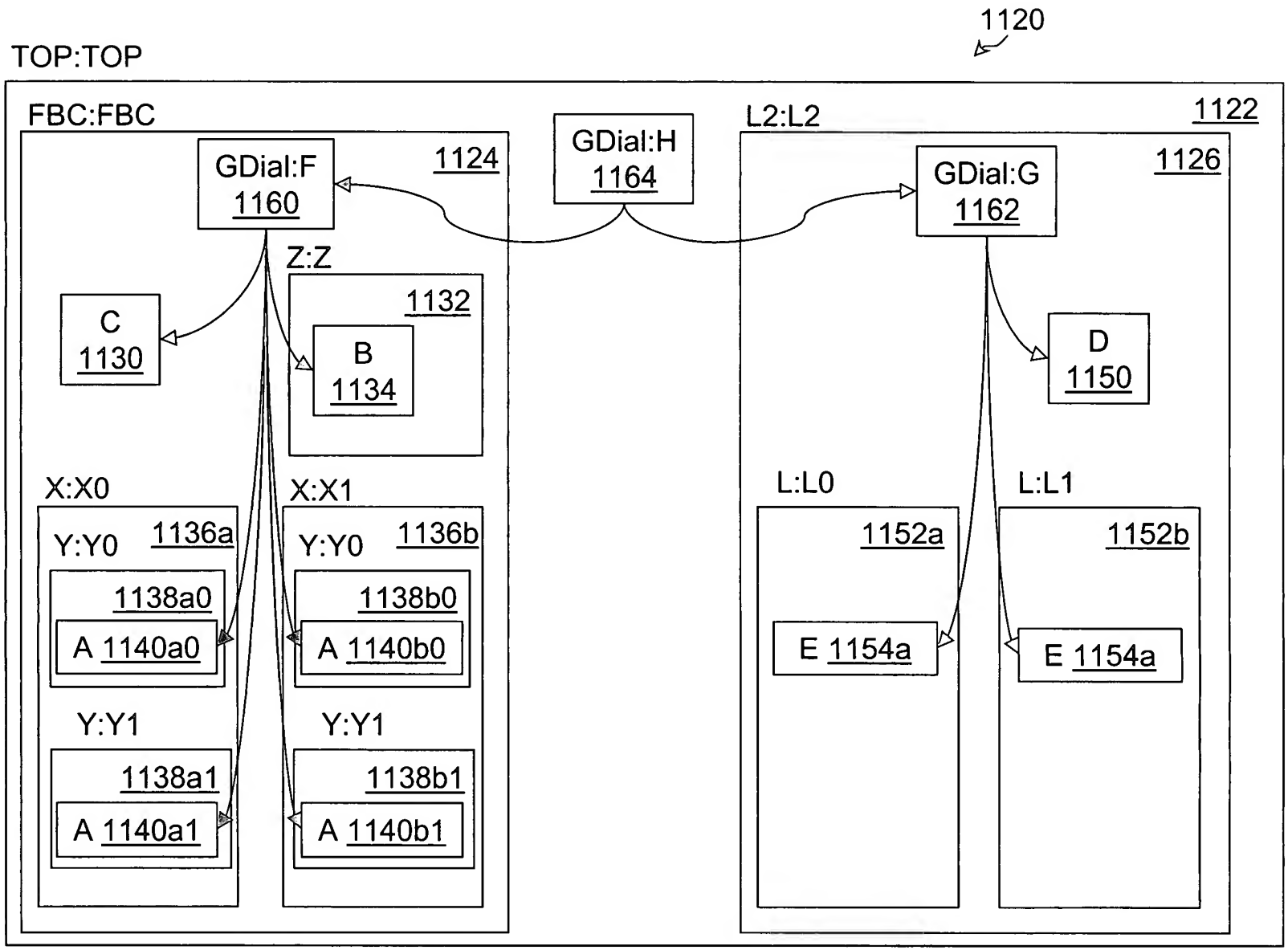
1120
TOP:TOP
FBC:FBC
GDial:F
1160
1124
GDial:H
1164
L2:L2
1122
GDial:G
1162
1126
Z:Z
1132
C
1130
B
1134
D
1150
X:X0
X:X1
L:L0
L:L1
Y:Y0
1136a
Y:Y0
1136b
1138a0
A 1140a0
1138b0
A 1140b0
1152a
1152b
E 1154a
E 1154a
Y:Y1
Y:Y1
1138a1
A 1140a1
1138b1
A 1140b1

Figure 11B

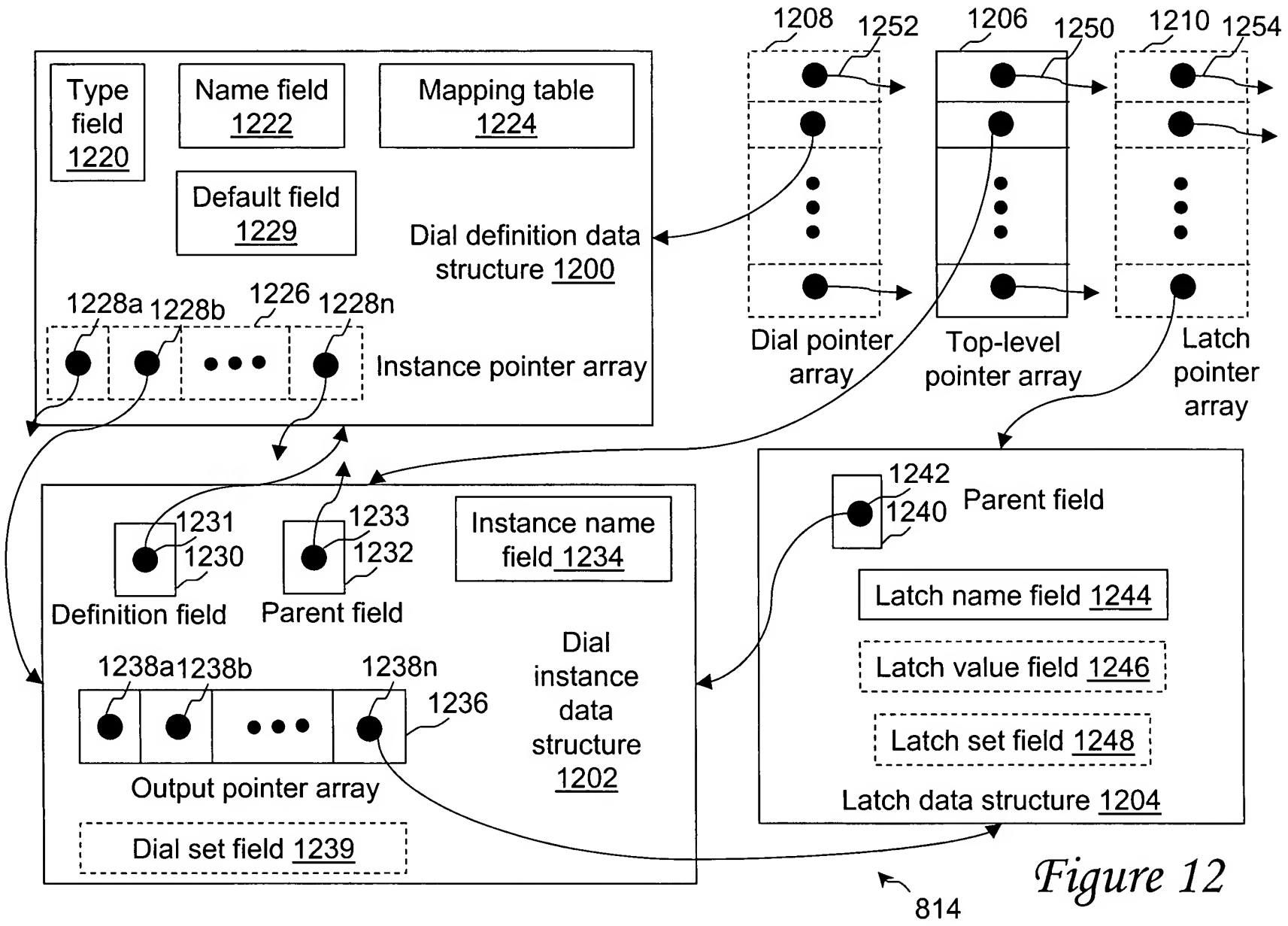
1208
1252
1206
1250
1210
1254
Type field 1220
Name field 1222
Mapping table 1224
Default field 1229
Dial definition data structure 1200
1228a
1228b
1226
1228n
Instance pointer array
Dial pointer array
Top-level pointer array
Latch pointer array
1242
1240
Parent field
Latch name field 1244
Latch value field 1246
Latch set field 1248
Latch data structure 1204
1231
1230
Definition field
1233
1232
Parent field
Instance name field 1234
1238a
1238b
1238n
1236
Output pointer array
Dial instance data structure 1202
Dial set field 1239
814

Figure 12

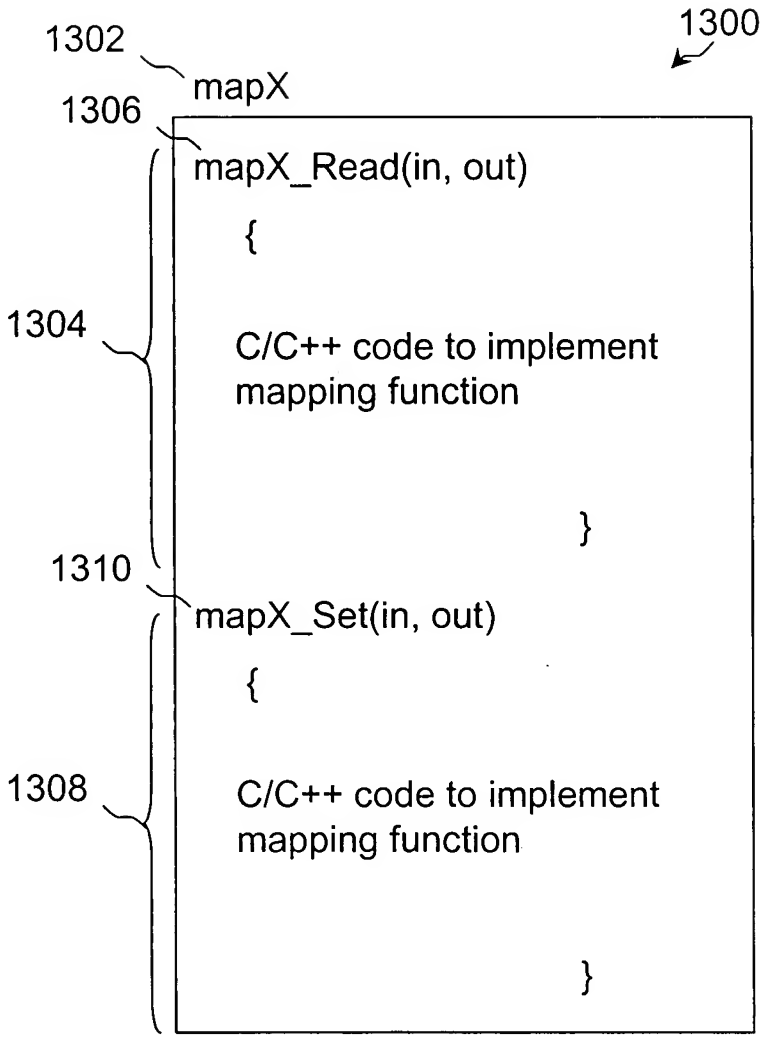

Figure 13A

1320

Mapping Control File

Mapping function name 1322	Read Input width 1324	Read Output width 1326	Set Input width 1328	Set Output width 1330
⋮	⋮	⋮	⋮	⋮

Figure 13B

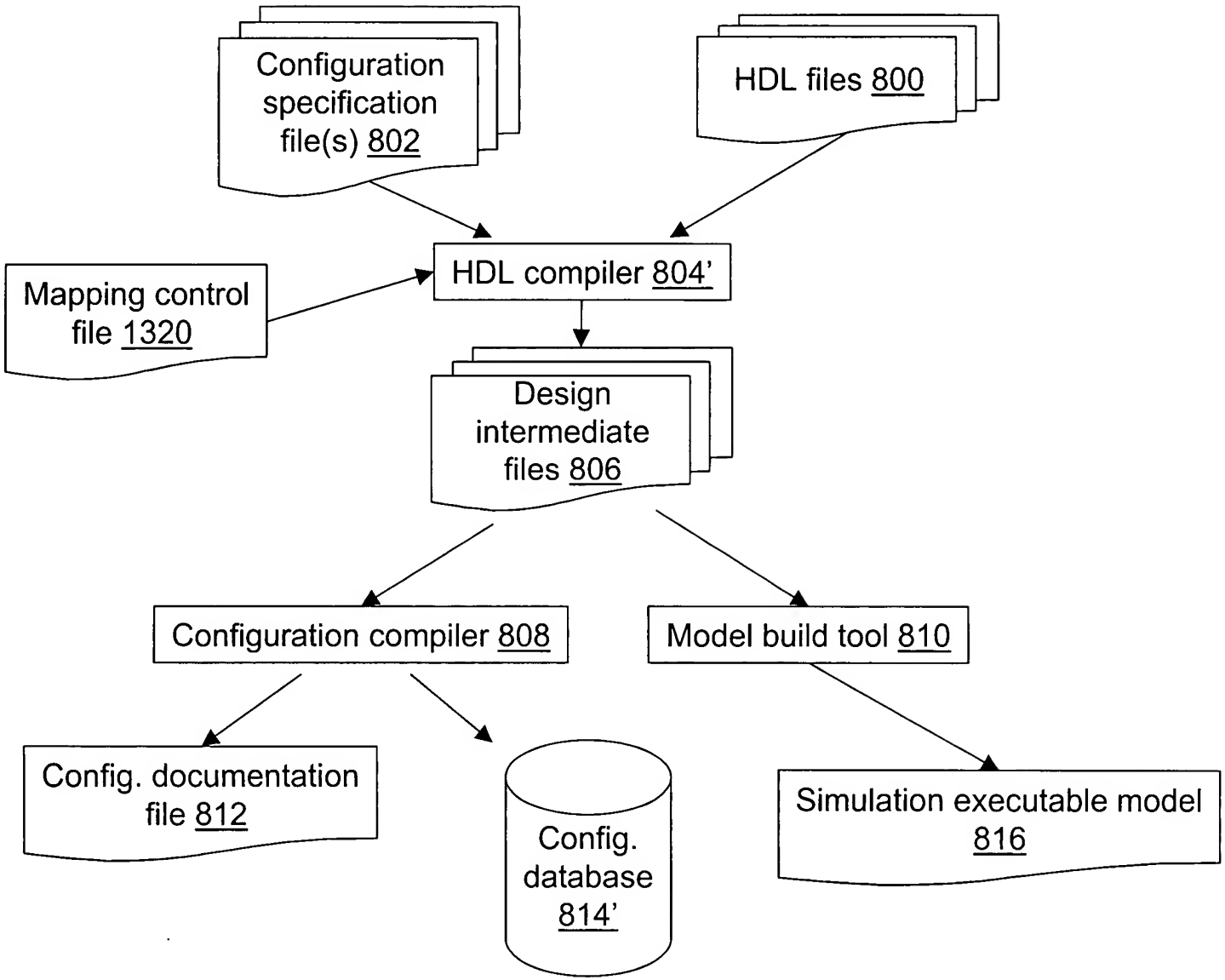
Configuration specification file(s) 802
HDL files 800
Mapping control file 1320
HDL compiler 804'
Design intermediate files 806
Configuration compiler 808
Model build tool 810
Config. documentation file 812
Config. database 814'
Simulation executable model 816

Figure 14

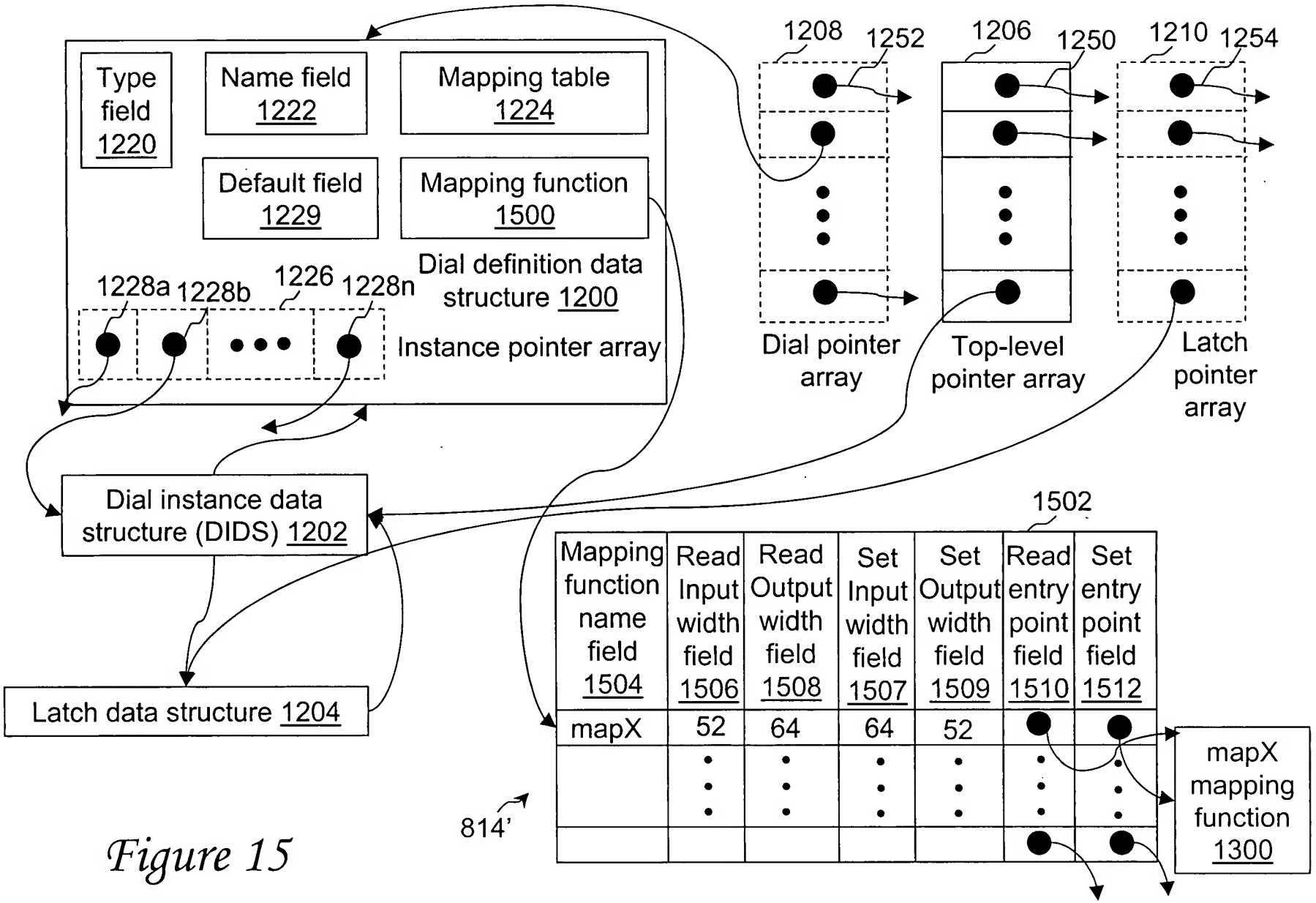
Type field 1220
Name field 1222
Mapping table 1224
Default field 1229
Mapping function 1500
Dial definition data structure 1200
1228a
1228b
1226
1228n
Instance pointer array
1208
1252
1206
1250
1210
1254
Dial pointer array
Top-level pointer array
Latch pointer array
Dial instance data structure (DIDS) 1202
Latch data structure 1204
1502
Mapping function name field 1504
Read Input width field 1506
Read Output width field 1508
Set Input width field 1507
Set Output width field 1509
Read entry point field 1510
Set entry point field 1512
mapX
52
64
64
52
814'
mapX mapping function 1300

Figure 15

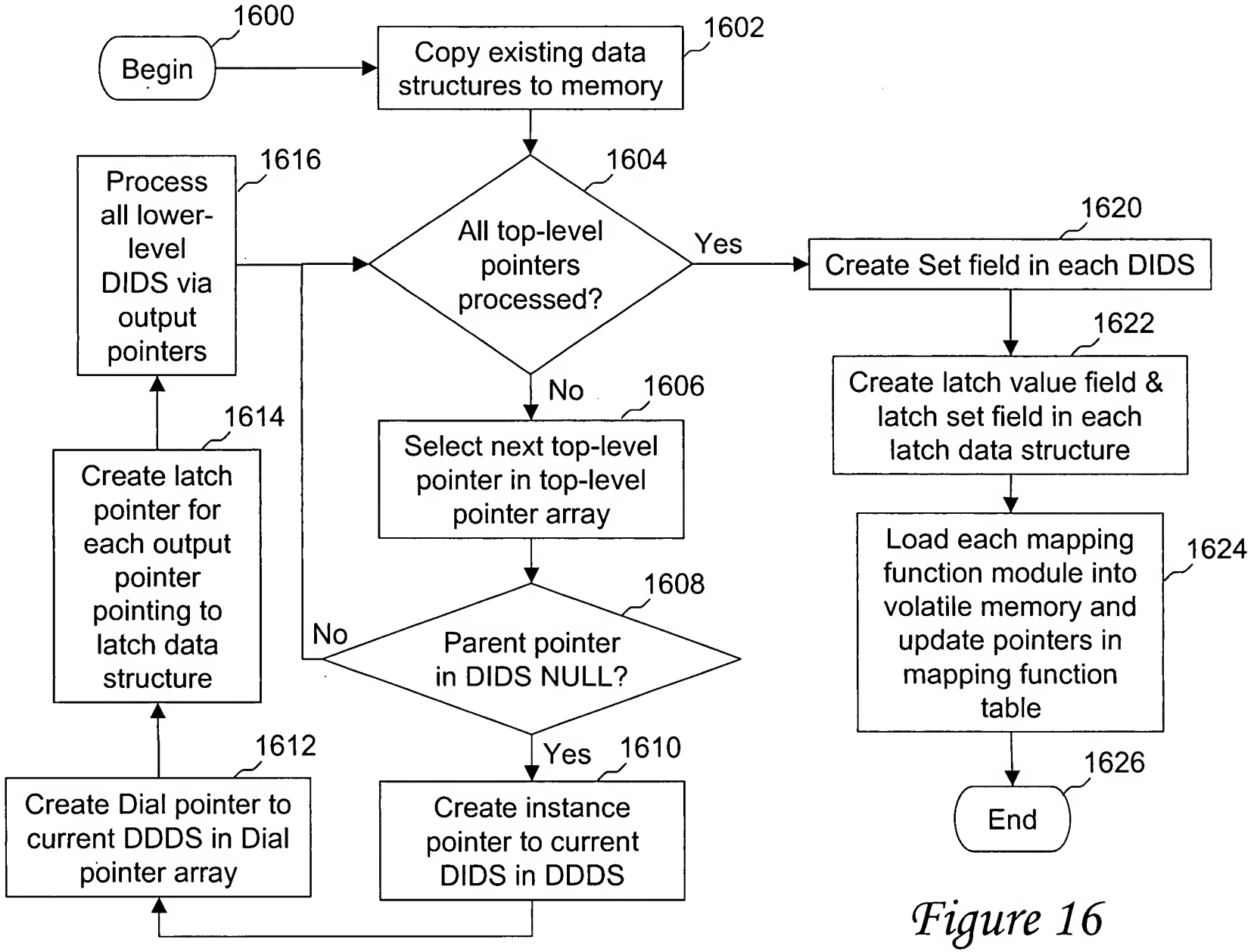
1600
Begin
1602
Copy existing data structures to memory
1604
All top-level pointers processed?
Yes
1620
Create Set field in each DIDS
1622
Create latch value field & latch set field in each latch data structure
1624
Load each mapping function module into volatile memory and update pointers in mapping function table
1626
End
No
1606
Select next top-level pointer in top-level pointer array
1608
Parent pointer in DIDS NULL?
No
Yes
1610
Create instance pointer to current DIDS in DDDS
1612
Create Dial pointer to current DDDS in Dial pointer array
1614
Create latch pointer for each output pointer pointing to latch data structure
1616
Process all lower-level DIDS via output pointers

Figure 16

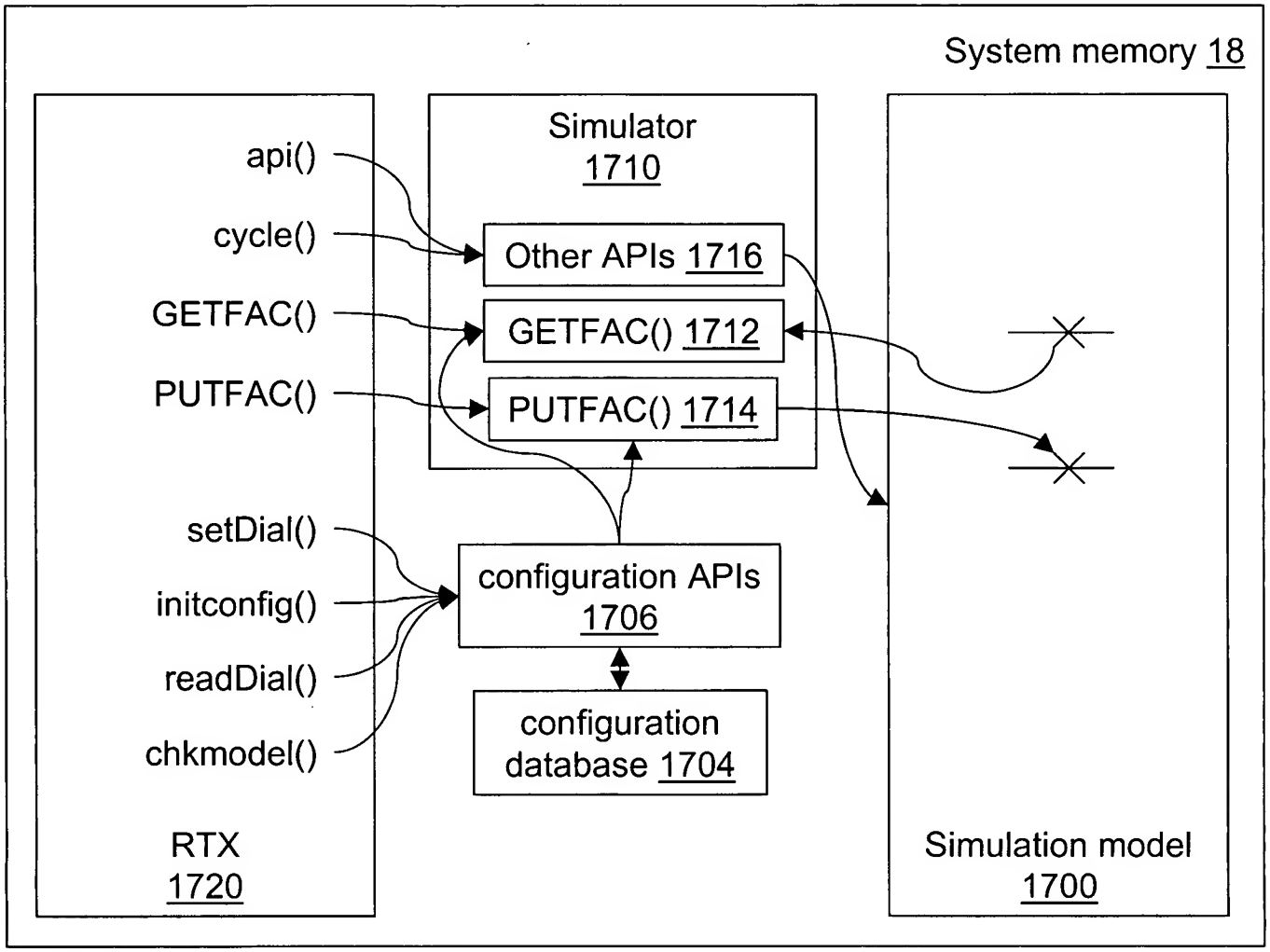
System memory 18
Simulator
1710
api()
cycle()
GETFAC()
PUTFAC()
Other APIs 1716
GETFAC() 1712
PUTFAC() 1714
setDial()
initconfig()
readDial()
chkmodel()
configuration APIs
1706
configuration
database 1704
RTX
1720
Simulation model
1700

Figure 17

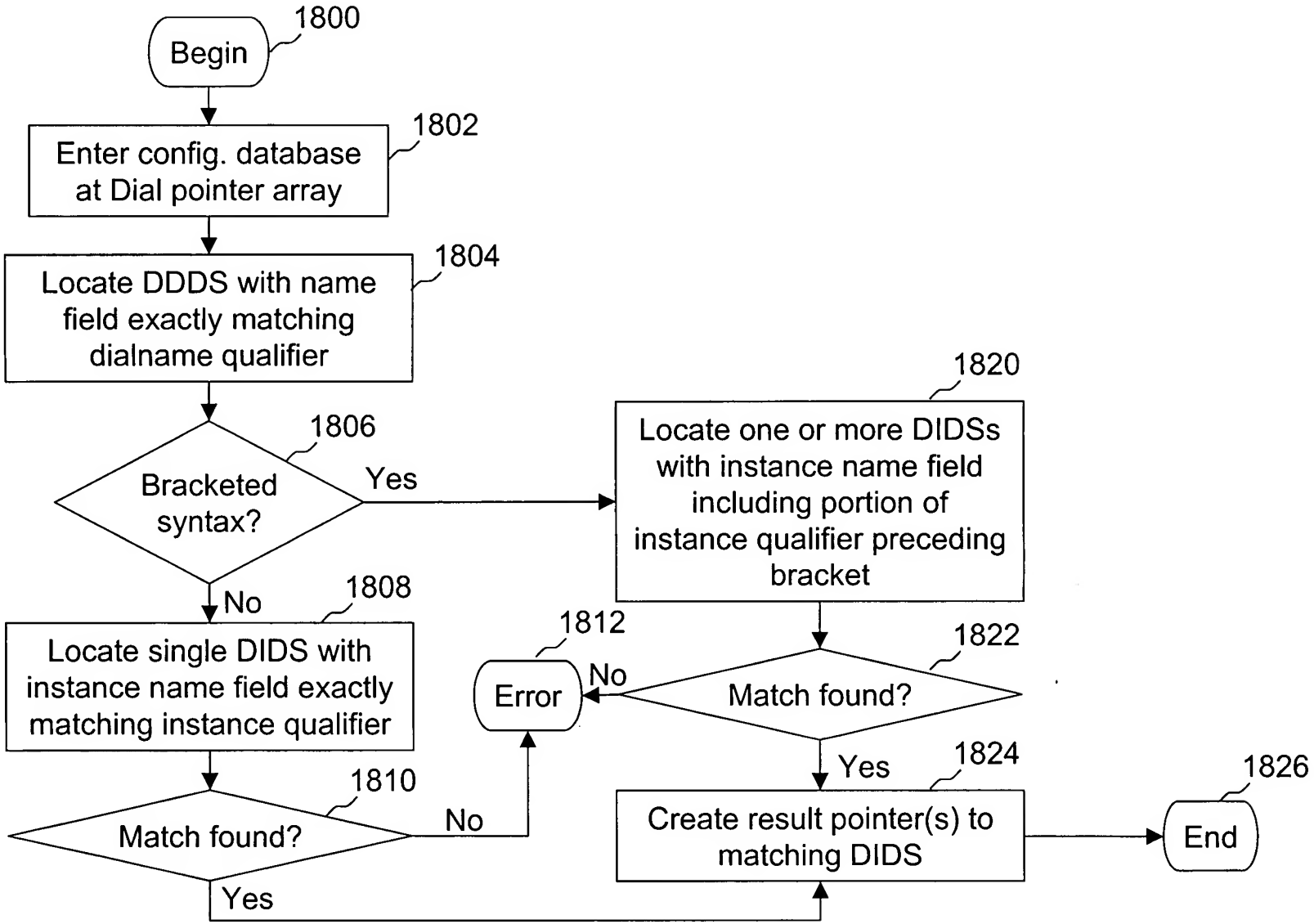
1800
Begin
1802
Enter config. database at Dial pointer array
1804
Locate DDDS with name field exactly matching dialname qualifier
1806
Bracketed syntax?
Yes
No
1808
Locate single DIDS with instance name field exactly matching instance qualifier
1810
Match found?
No
Yes
1812
Error
1820
Locate one or more DIDSs with instance name field including portion of instance qualifier preceding bracket
1822
Match found?
No
Yes
1824
Create result pointer(s) to matching DIDS
1826
End

Figure 18

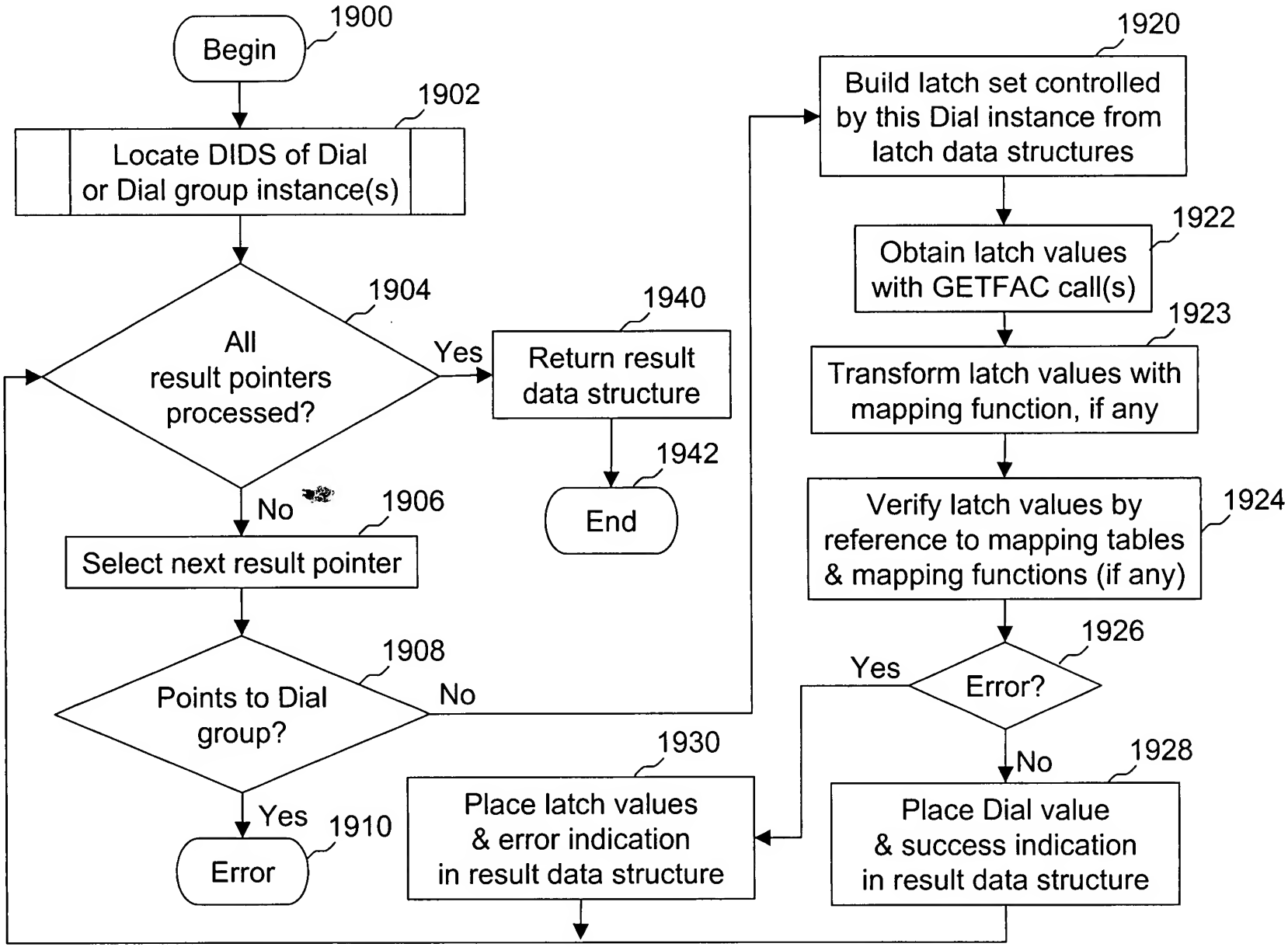
Begin
1900
Locate DIDS of Dial or Dial group instance(s)
1902
All result pointers processed?
1904
Yes
Return result data structure
1940
End
1942
No
Select next result pointer
1906
Points to Dial group?
1908
No
Yes
Error
1910
Build latch set controlled by this Dial instance from latch data structures
1920
Obtain latch values with GETFAC call(s)
1922
Transform latch values with mapping function, if any
1923
Verify latch values by reference to mapping tables & mapping functions (if any)
1924
Error?
1926
Yes
No
Place latch values & error indication in result data structure
1930
Place Dial value & success indication in result data structure
1928

Figure 19A

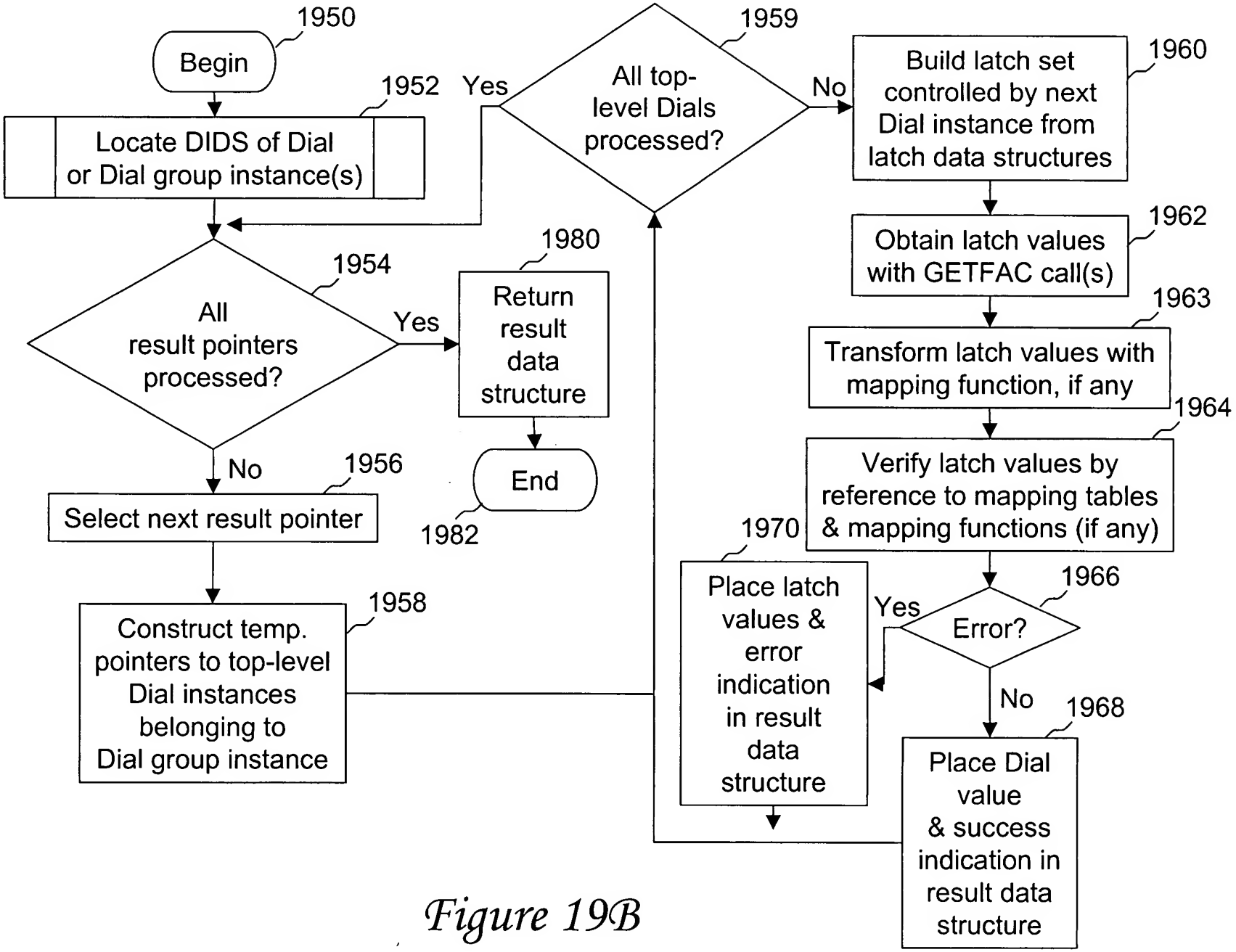
1950
Begin
1952
Locate DIDS of Dial or Dial group instance(s)
1954
All result pointers processed?
Yes
1980
Return result data structure
End
1982
No
1956
Select next result pointer
1958
Construct temp. pointers to top-level Dial instances belonging to Dial group instance
1959
All top-level Dials processed?
Yes
No
1960
Build latch set controlled by next Dial instance from latch data structures
1962
Obtain latch values with GETFAC call(s)
1963
Transform latch values with mapping function, if any
1964
Verify latch values by reference to mapping tables & mapping functions (if any)
1966
Error?
Yes
1970
Place latch values & error indication in result data structure
No
1968
Place Dial value & success indication in result data structure

Figure 19B

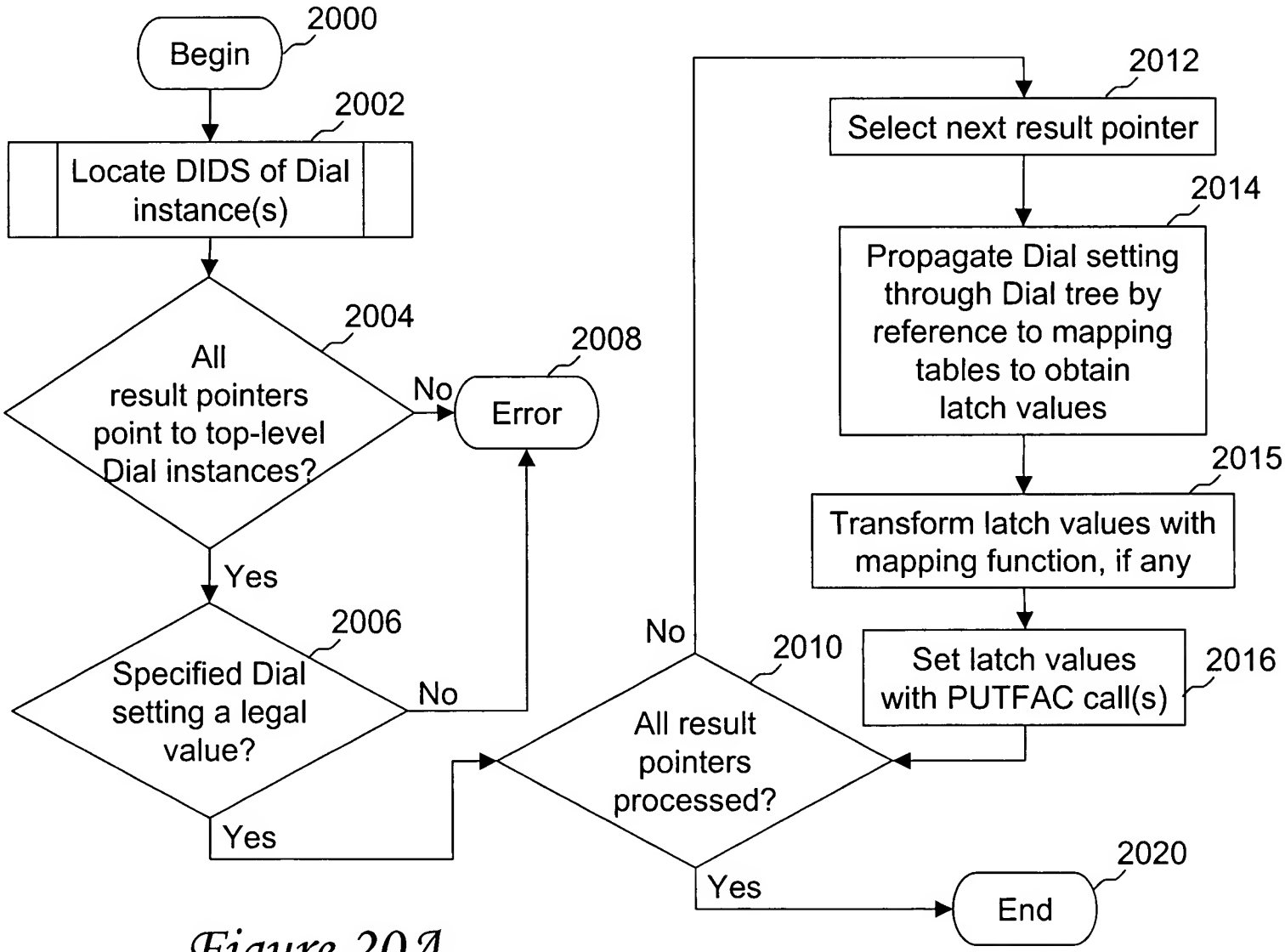
Begin
2000
Locate DIDS of Dial instance(s)
2002
All result pointers point to top-level Dial instances?
2004
No
Error
2008
Yes
Specified Dial setting a legal value?
2006
No
Yes
All result pointers processed?
2010
No
Select next result pointer
2012
Propagate Dial setting through Dial tree by reference to mapping tables to obtain latch values
2014
Transform latch values with mapping function, if any
2015
Set latch values with PUTFAC call(s)
2016
Yes
End
2020

Figure 20A

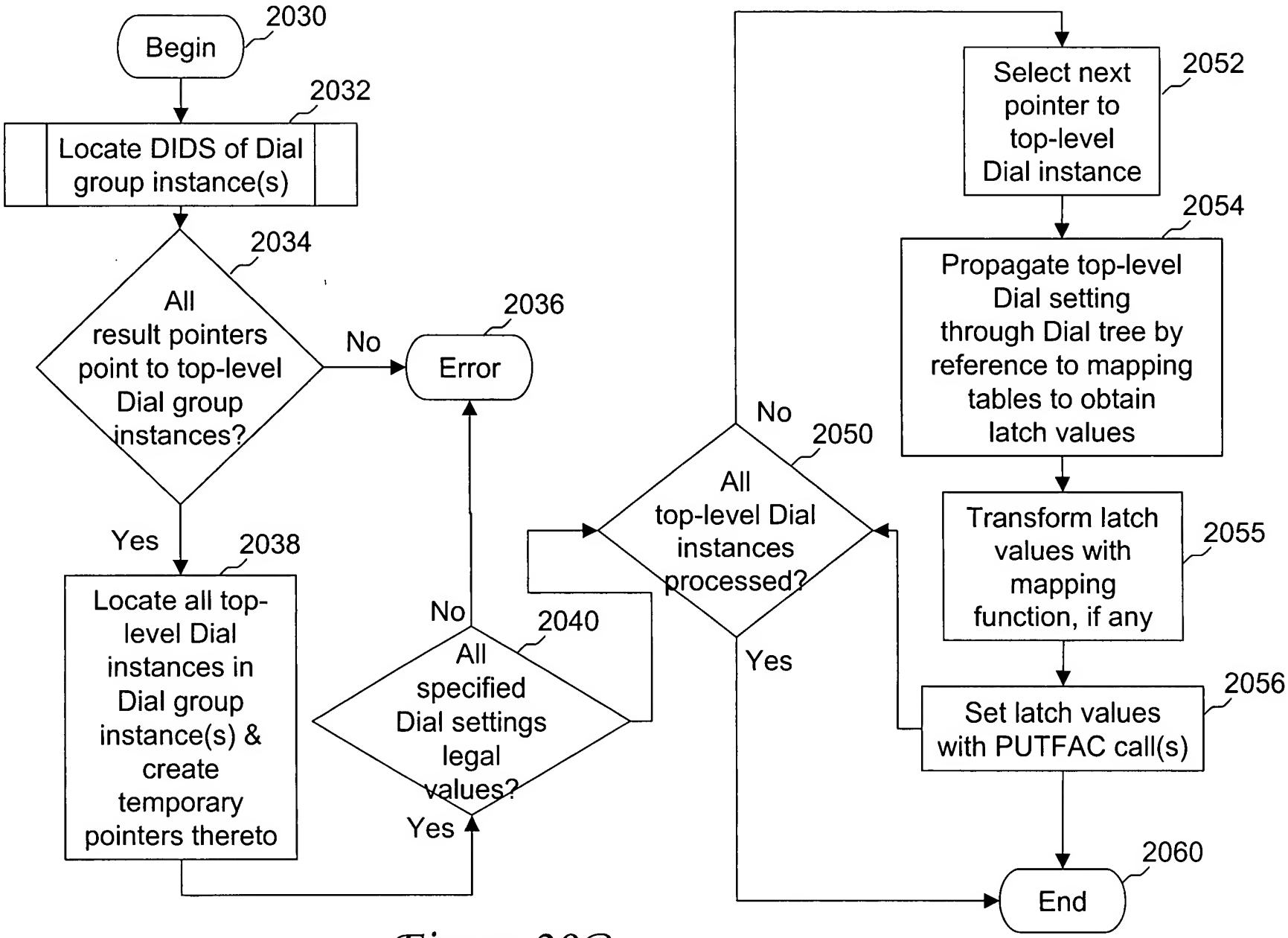
Begin
2030
Locate DIDS of Dial group instance(s)
2032
All result pointers point to top-level Dial group instances?
2034
No
Error
2036
Yes
Locate all top-level Dial instances in Dial group instance(s) & create temporary pointers thereto
2038
All specified Dial settings legal values?
2040
No
Yes
All top-level Dial instances processed?
2050
No
Yes
Select next pointer to top-level Dial instance
2052
Propagate top-level Dial setting through Dial tree by reference to mapping tables to obtain latch values
2054
Transform latch values with mapping function, if any
2055
Set latch values with PUTFAC call(s)
2056
End
2060

Figure 20B

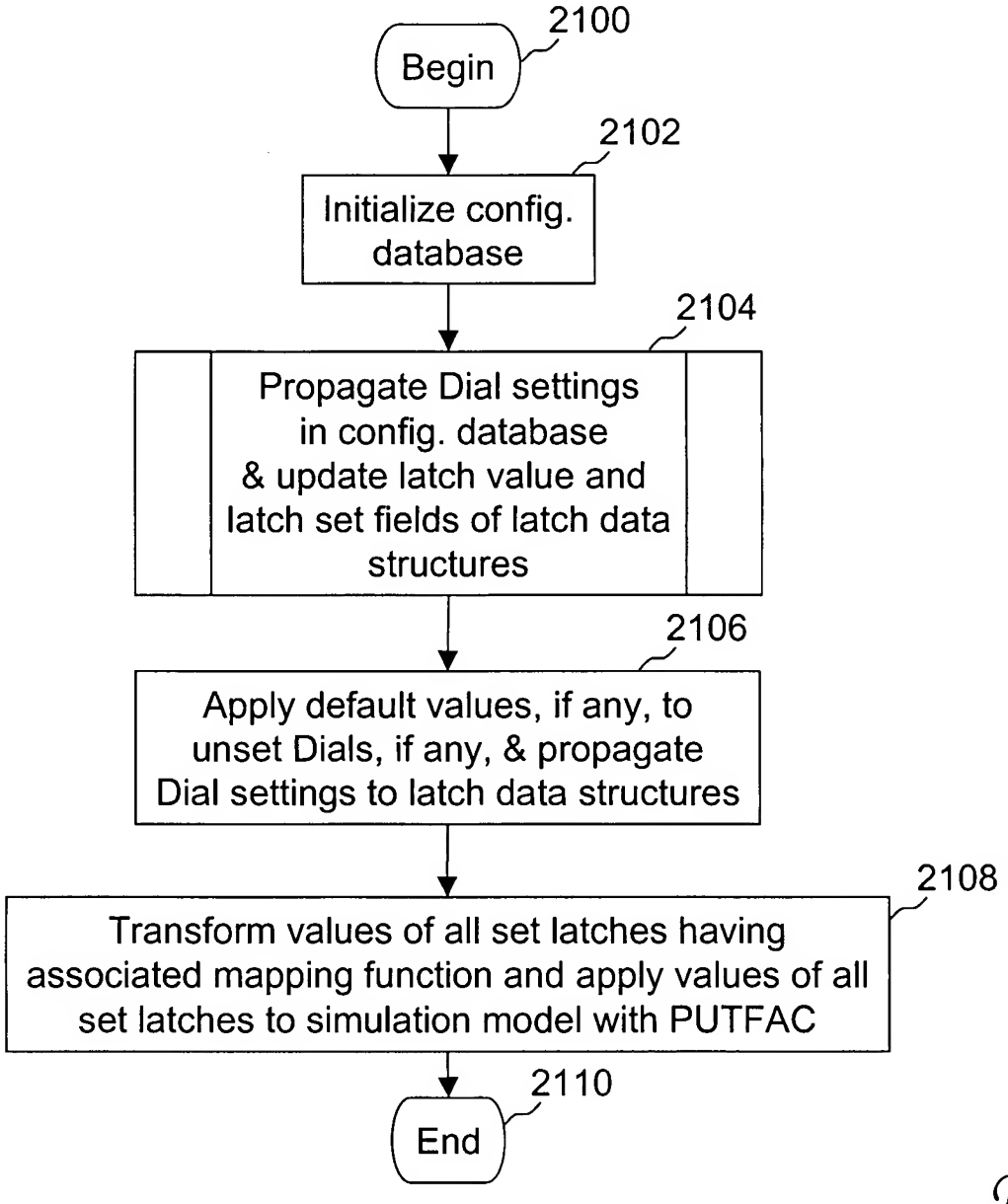
2100
Begin
2102
Initialize config. database
2104
Propagate Dial settings in config. database & update latch value and latch set fields of latch data structures
2106
Apply default values, if any, to unset Dials, if any, & propagate Dial settings to latch data structures
2108
Transform values of all set latches having associated mapping function and apply values of all set latches to simulation model with PUTFAC
2110
End

Figure 21